DIGITAL INTERPOLATION

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

by

RASHID ANSARI

to the

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

AUGUST 1977

CERTIFICATE

This is to certify that the thesis entitled 'Digital Interpolation' is a record of the work carried out under my supervision and that it has not been submitted elsewhere for a degree.

Dr. S.K. Mullick
Professor
Department of Electrical Engineering
Indian Institute of Technology
Kanpur

ACKNOWLEDGEMENT

I am highly obliged to Prof. S.K. Mullick for his guidance, frequent discussions and helpful suggestions throughout the course of this work. The informal working relationship went a long way in making the work enjoyable.

I would like to record my appreciation for Mr. K.N. Tewari's efficiently typing of this thesis.

Kanpur,
August,1977

Rashid Ansari

CONTENTS

ABSTRACT

This thesis deals with the problem of Digital Interpolation, or the process of increasing the sampling rate of a sampled data signal. It is mainly concerned with investigating the possibility of improving the performance of interpolators for bandlimited signals regarding which no additional information is available. The approach suggested is to estimate the MEM spectrum and design optimum interpolators based on this estimate. Using this approach, extensive studies on a representative set of signals were carried out. A comparative evaluation of the performance of the various interpolation schemes was made on the basis of minimum mean square error criteria. The degree of improvement resulting from the suggested method is shown to depend on the nature of the spectrum. The thesis also includes a review of work in Digital Interpolation, various interpretations and approaches to the problem and useful computer programs used in the study.

CHAPTER 1

INTRODUCTION

In digital signal processing there is often a need to alter the sampling rate of signals. Digital Interpolation refers to the process of increasing the sampling frequency by filling in more samples in between those available in the input data. A related process, one of decreasing the sampling rate, is referred to as decimation. In most applications, the two operations are employed in a complementary manner.

At the outset, the distinction between interpolation as is to be understood in digital processing and its usual concept in numerical analysis should be made clear. In the latter, interpolation is normally performed by constructing a polynomial fit of an adequate order to the given data. Often a linear interpolation suffices. A study of the process in the frequency domain shows that these methods are not the most suitable schemes for interpolation (1). For the class of bandlimited signals, interpolation is shown to be essentially a low pass filtering process. Thus the objective in the design of an interpolator is to obtain the best approximations to an ideal low pass filter.

APPLICATIONS OF DIGITAL INTERPOLATION

It is instructive at this stage to look at some of the applications of interpolation. Of the several applications, we choose to highlight a few.

(a) Vocoding:

In this method of data compression, information about slowly varying speech parameters obtained by the analysis of natural utterances is transmitted at low sampling rates. The regeneration of speech requires a higher sampling rate for synthesis (2), (3). This is achieved by interpolation.

(b) Code Format Conversion:

For Delta Modulation (DM) and Pulse Code Modulation (PCM), the required sampling rates are inherently different. A filter that effects a conversion from DM to PCM is basically a rate changing filter (4),(5).

(c) Digital Realisation of Frequency Division Multiplexing (FDM);

In an efficient realisation of an FDM system employing digital filtering, complicated operations are performed at low sampling rates before translation to higher rates for grouping of channels in the final stage (6). Here we have a need for interpolation at the transmitter and decimation at the receiver.

(d) Two-dimensional Processing:

Sometimes the array of digital data in the sampled version of a picture may not be adequate for a good display. Interpolation helps put the picture into better evidence. On other occasions, for instance, in the data collected by satellite observation of the earth, the grid points in the input data represent a distorted perspective projection of the object of observation. Geometrical correction on the entire data requires excessive computations. Instead the correlation is performed on a sparse input grid and interpolation is carried out on the corrected output grid.(32)

(e) Efficient Filter Realisations:

Interpolation techniques are now being put to wide use in improving computational efficiencies of Digital [illegible]. Filters. The feature that is exploited here is that [illegible] computations performed at a rate equal to twice the highest output frequency are adequate. If the sampling rate of the input is high, the data can be decimated and filtered before being restored to the original sampling rate by interpolation. A great deal of investigation has been carried out on multirate filtering (7)-(12).

REVIEW OF RESEARCH IN DIGITAL INTERPOLATION

If we trace the developments in this field, we recognise the paper by Schafer and Rabiner in 1973 (1) as a landmark publication. It put the problem in the

right perspective from the point of view of digital filtering. The authors suggested methods for obtaining change in sampling rates with rational ratios. A strong case was built up for the use of Finite Impulse Response (FIR) filters. It was soon recognised that when the initial and final rates differ widely, using a sequence of filters proves more efficient than using a single stage filter. Using this approach, a multistage procedure for rate reduction in the ratios of 2:1 was proposed (5),(11),(13). Here the use of halfband filters considerably reduces the computations. Recently a design procedure for properly choosing a sequence of half band filters that meets specified fidelity criteria, has been presented (14).

Without imposing conditions on the type of FIR filters, a cascading procedure wherein the sampling rate reduction at each stage can be greater than 2 has been suggested in (15). The attempt is to optimally allocate the decimation to minimise the overall amount of computation. This approach has been extended to the general case of multistage processing (16). The problem of designing a multistage interpolator for minimisation of storage rather than computation has also been examined (17).

For interpolation, the use of Infinite Impulse Response (IIR) filters in the conventional form has been

shown to be less efficient than the use of FIR filters (1). But use of IIR filters in phase-shifting networks for obtaining rate alterations in an efficient way has been proposed in (18). Another method utilizes a state variable description to produce realisations of sample rate reduction filters (10).

Digital Interpolation has been analysed, in the time domain also and the analysis enables us to optimally design a filter for minimisation of error (19),(20). In the optimisation, stationarity together with a knowledge of autocorrelation function is assumed.

Another method of interpolation employs the Discrete Fourier Transform approach (21). For this method, interpolation kernels and expressions for error bounds have been derived.

SCOPE OF THE PRESENT WORK

The present work is concerned mostly with the comparative evaluation of various interpolation schemes using a representative set of signals and a performance measure based on mean square error. A new method of interpolation by choosing filter coefficients on the basis of the received data has been tried. Without assuming a-priori knowledge of the autocorrelation function, an attempt is made by using the Maximum Entropy Method (MEM) to estimate

the Autocorrelation function. This estimate in turn is then used to design an optimum filter. Also a direct operation on the time sequence data using an MEM approach has been tried.

After a preliminary discussion on the interpretation of interpolation and interpolation schemes in Chapter 2, the suggested approach is elucidated in Chapter 3. A set of representative signals were used to test the efficacy of the interpolation techniques discussed in Chapters 2 and 3 and these results are presented in Chapter 4. Chapter 5 analyses the results and makes recommendations regarding the application of the various schemes for interpolation. Listings of some of the programms used in these studies are included in the appendix.

CHAPTER 2

AN ANALYTICAL INSIGHT INTO THE INTERPOLATION PROCESS

It should be pointed out right at the beginning that we shall be concerned here only with linear interpolation. In this chapter, we first take a brief look at the classical techniques of interpolation. We next develop a frequency domain interpretation of digital interpolation and discuss design criteria for interpolators. A result that emerges from the discussion is that FIR filters are particularly suited for interpolation. We proceed to discuss methods for efficient implementation of the interpolation schemes. A novel IIR implementation described in (18), based on phase shifting networks, is also examined.

In the latter half of the chapter, the problem is formulated in the time domain. The optimum filter for a class of signals with a known autocorrelation function is derived.

2.1. Classical Interpolation Techniques

A vast amount of literature is available on interpolation techniques used in numerical analysis. We restrict our attention to aspects that are relevant to digital interpolation.

Suppose that the values of a continuous function $x'(t)$ are available at equispaced points in time. That is, a sequence $x(n)$ is available to us where

$$x(n) = x'(nT) \tag{2.1.1}$$

The requirement is to obtain a set of values at a different time spacing $T' = T/L$, L integer. In other words we are required to estimate a sequence $y(n)$ from the available data such that $y(n)$ approximates $x'(nT')$. In numerical analysis, this objective is realised by constructing a polynomial of degree P which passes through (P+1) observations around the point of interpolation. The order P is chosen according to the accuracy desired. Clearly this interpolating polynomial is unique. A convenient method of computation is to construct difference tables of various types like forward, backward and central differences (26). Based on these difference schemes, a host of interpolation techniques like Lagrange, Gauss, Stirling, Bessel and Everctt have evolved. The form that is most amenable to digital interpolation is the Lagrange form. For the pth interpolation, $y(mL+p)$, between $x(m)$ and $x(m+1)$, let us use $(K+K'+1)$ samples $x(m-K), \ldots, x(m), \ldots, x(m+K')$. Then

$$y(mL+p) = \sum_{k=-K}^{K'} L_k(mL+p)\, x(m+k) \tag{2.1.2}$$

where $p = 0, 1, \ldots, L-1$ and $L_k(.)$, the Lagrange interpolating polynomial

$$L_k(x) = \frac{(x-x_1)\ldots(x-x_{k-1})(x-x_{k+1})\ \ldots\ (x-x_n)}{(x_k-x_1)\ \ldots\ (x_k-x_{k-1})(x_k-x_{k+1})\ldots(x_k-x_n)}$$

The polynomial as adapted to discrete time, is given by

$$L_k(mL+p) = \frac{(K+p/L)(K-1+p/L)\ldots(-(k-1)+p/L)(-(k+1)+p/L)\ldots(-K'+p/L)}{(k+K)(k+K-1)\ \ldots\ (1)\ (-1)\ \ldots\ (k-K')} \qquad (2.1.3)$$

Clearly for $p = 0$

$$L_k(mL) = \delta_{ok}$$

which means that $y(mL) = x(m)$. The input samples are thus preserved. The right hand side of (2.1.3) is independent of m. Hence it is evident that (2.1.2) can be implemented by a convolution operation using a FIR filter whose coefficients are determined by (2.1.3).

2.2 Frequency Domain Interpretation

The analysis presented here is based on (1). As in (2.1.1) we consider a sampled sequence $x(n)$

$$x(n) = x'(nT) \qquad (2.2.1)$$

The fourier transform $X'(w)$ of $x'(t)$ is

$$X'(w) = \int_{-\infty}^{\infty} x'(t)\ e^{-jwt}\ dt$$

The z-Transform of the sequence $x(n)$ is

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n} \qquad (2.2.2)$$

$X(e^{jwT})$, the Fourier transform of the sequence $x(n)$ is related to $X'(w)$, (22), as follows:

$$X(e^{jwT}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X'(w + \frac{k2\pi}{T}) \qquad (2.2.3)$$

Consider $x'(t)$ to be bandlimited, i.e.

$$X'(w) = 0 \text{ for } |w| \geq B$$

as shown in Figure 1. If $T \leq \pi/B$, then from (2.2.3) it is seen that

$$X(e^{jwT}) = \frac{1}{T} X'(w), \; -\pi/T \leq w \leq \pi/T \qquad (2.2.4)$$

It is well known that the Sinc functions defined as

$$\mathrm{Sinc}(t/T) = \frac{\mathrm{Sin}(\pi t/T)}{(\pi t/T)} \qquad (2.2.5)$$

can be used for interpolation. But such interpolation is not practicable as the Sinc functions have an infinite duration with significant contribution even at points widely separated. Interpolation functions which decay more rapidly than the Sinc functions, like

$$I(t/T) = \frac{\mathrm{Sin}(\pi t/T)}{\mathrm{Sinh}(\pi t/T)} \qquad (2.2.6)$$

have been suggested (27). For the purpose of implementation these functions have to be truncated. Instead of using such truncated functions, it is more reasonable to look for finite

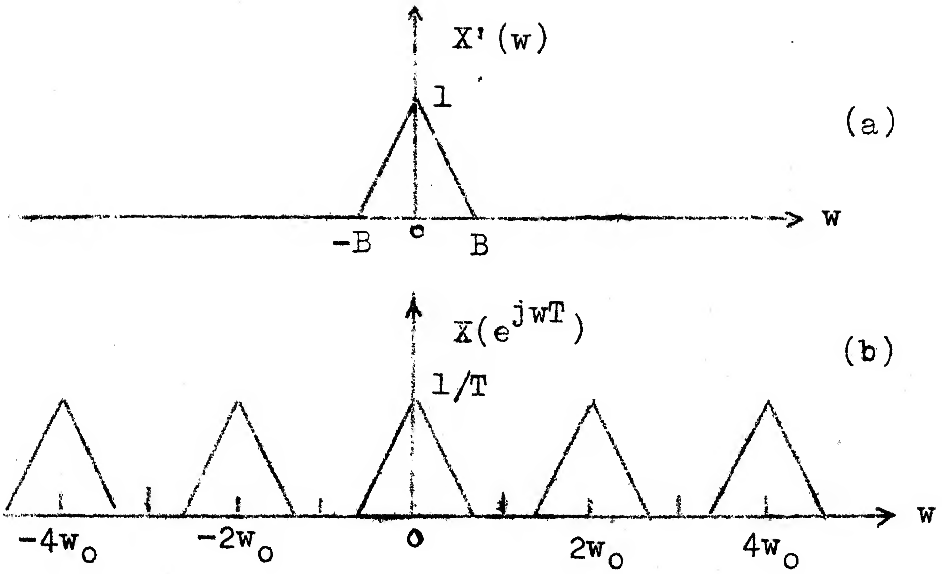


Figure 2.1: (a) Fourier Transform of a continuous time signal.

(b) Fourier Transform of the sequence obtained by sampling with a period $T \leq \pi/B$. ($w_o = \pi/T$).

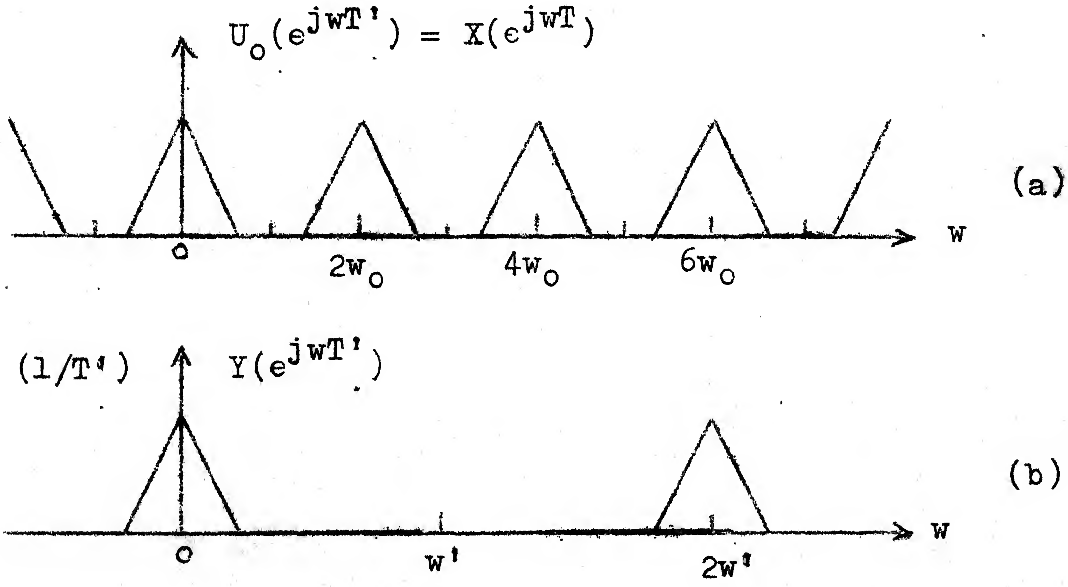


Figure 2.2: Sampling rate increase ($T' = T/3$, $w_o = \pi/T$, $w' = \pi/T'$).

(a) Fourier Transform of sequences $u_o(n)$ and $x(n)$.

(b) Fourier Transform of interpolated sequence $y(n)$.

duration functions. The search is facilitated by the analysis that follows:

2.1(a) Sample Rate Increase:

Suppose it is desired to increase the sampling rate by an integer factor L. The new sampling period is

$$T' = T/L$$

Here we seek to fill in (L-1) samples between each two adjacent samples of the input data. A new sequence $u_o(n)$ is defined as

$$u_o(n) = x(n/L),\ n = mL \text{ and } m = 0, \pm 1, \pm 2, \ldots$$
$$= 0 \quad , \quad \text{otherwise} \qquad (2.2.7)$$

The z-Transform of the sequence $u_o(n)$ is

$$U_o(z) = \sum_{n=-\infty}^{\infty} u_o(n) z^{-n}$$

$$= \sum_{n=-\infty}^{\infty} x(n) z^{-Ln}$$

$$= X(z^L) \qquad (2.2.8)$$

The Fourier transform of the sequence $u_o(n)$ is

$$U_o(e^{jwT'}) = X(e^{jwT'L})$$

$$= X(e^{jwT}) \qquad (2.2.9)$$

$U_o(e^{jwT'})$ thus has a period $2\pi/T$ rather than $2\pi/T'$ as is

normally the case for signals sampled at a period T'.

The desired sequence $y(n)$ is such that

$$y(n) = x'(nT')$$

This is achieved if we ensure that

$$Y(e^{jwT'}) = \frac{1}{T'} X'(w), \quad -\pi/T' \leq w \leq \pi/T' \qquad (2.2.10)$$

The required filter should produce the sequence $y(n)$ by operating on $u_o(n)$. In the Transform domain

$$Y(e^{jwT'}) = H(e^{jwT'})\, U_o(e^{jwT'})$$

$$= H(e^{jwT'})\, X(e^{jwT}) \qquad (2.2.11)$$

That is, $Y(e^{jwT'}) = \frac{1}{T} H(e^{jwT'})\, X'(w), \quad -\pi/T \leq w \leq \pi/T$

(2.2.12)

The implication of the above equations is illustrated in Figure 2.2. The requirement is to remove the $(L-1)$ images of $(1/T)X'(w)$ in $U_o(e^{jwT'})$ which lie between $w = \pi/T$ and $w = (2L-1)\pi/T$ and which are centred at $w = 2\pi/T, 4\pi/T, \ldots, 2(L-1)\pi/T$. A digital lowpass filter that rejects all frequencies in the range $\pi/T \leq w \leq \pi/T'$ meets the above requirements. From (2.2.10) and (2.2.12) we see that the passband gain must be L. Clearly, the ideal interpolator is

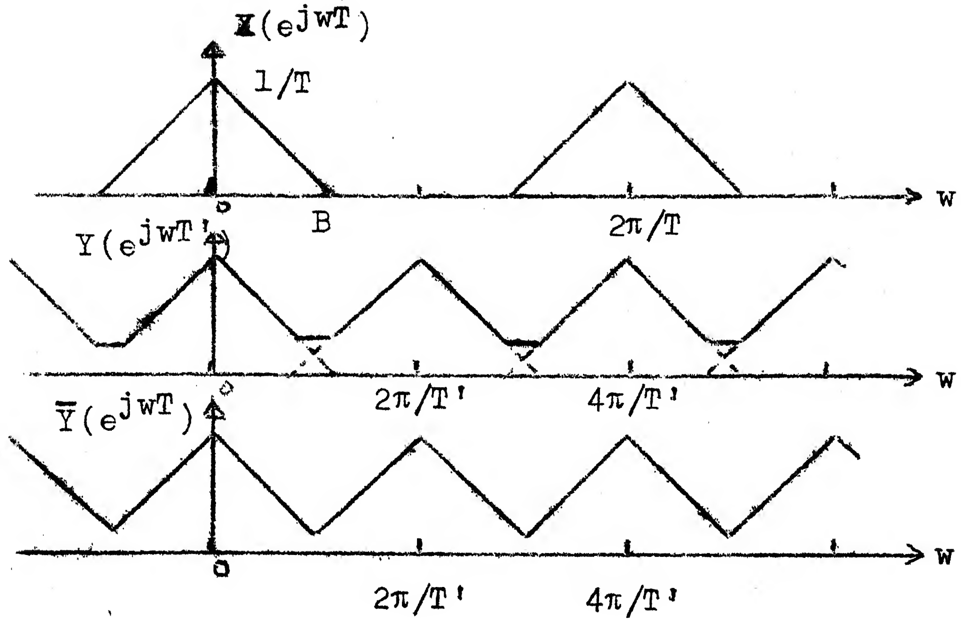


Figure 2.3: Sampling rate reduction (T'=2T)

(a) Fourier Transform of x(n), (b) Fourier Transform of y(n) with aliasing, (c) Fourier transform of y(n) obtained with a pre-cutoff.

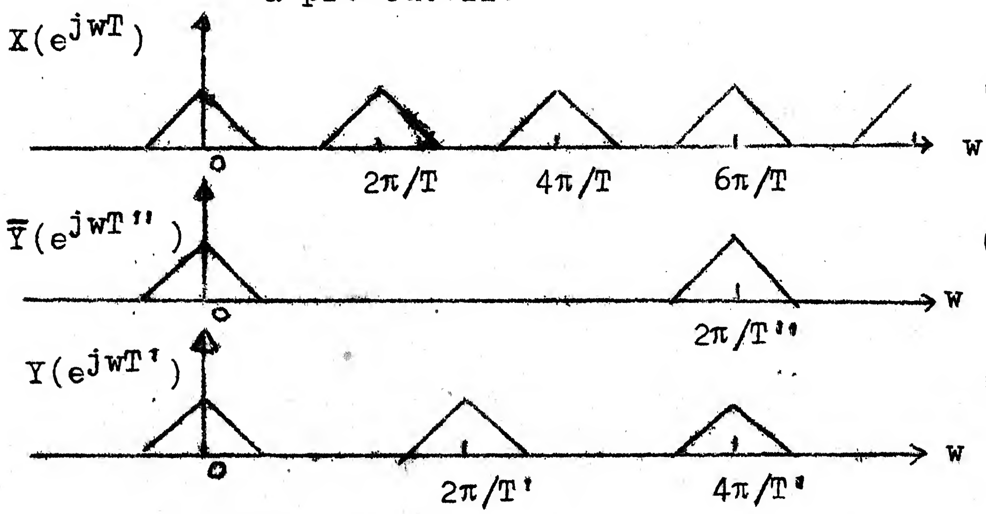


Figure 2.4: Sampling rate change by a factor 3/2.

(a) Fourier transform of x(n)

(b) Fourier Transform of interpolated intermediate samples,

(c) Fourier Transform of final sequence.

$$H(e^{jwT'}) = L \;, \quad |w| \leq \pi/T \qquad (2.2.13)$$
$$= 0 \;, \quad (\pi/T) < |w| \leq (\pi/T')$$

2.1(b) Sample Rate Decrease-Integer Factors:

If the sampling rate is decreased by an integer factor M, then the new sampling period is $T'=MT$. The required sequence is

$$y(n) = x(Mn)$$

We therefore retain one out of every M samples with a spacing of period T'. If the decimated samples are to uniquely determine $x'(t)$, the Sampling Theorem requires that $T' \leq \pi/B$. If this requirement is violated then aliasing errors occur. To avoid aliasing, lowpass filtering is performed before decimation.

A sampling rate reduction with M = 2 is illustrated in Figure 2.3 for the two cases of $T' < \pi/B$ and $T' > \pi/B$. Also shown is the case where, in order to avoid aliasing, the sequence has first been passed through a lowpass filter with a cut-off frequency π/T'.

2.1(c) Changing Sampling Rates by Non-integer Ratios:

We now consider changes in sampling rate by non-integer but rational factors (L/M). To achieve this, the above two techniques are combined by carrying out

interpolation by a factor L followed by decimation by a factor M. For the case $M > L$, if the new period T' is such that $T' = MT/L > \pi/B$, then aliasing has to be taken care of. Figure 2.4 illustrates a sampling rate increase by a factor 3/2.

2.3 Frequency Domain Design of Digital Interpolation Filters

It is clear from the frequency domain analysis that sampling rate alteration requires a lowpass filter. An obvious question that arises at this point is the type of the filter to be used. A basic consideration is the choice between FIR and IIR filters. Other considerations are regarding the specification of filters and the efficiency of implementation. These aspects are studied in this section.

2.3(a) The Case for FIR filters:

The ideal interpolator should have a zero phase or at most a linear phase. IIR filters do not possess this property. On the other hand several techniques are known for realising linear phase FIR filters (23), (24). Thus FIR filters eliminate errors due to phase non-linearity, and also provide arbitrarily small values of error due to amplitude distortion.

For conventional filters, IIR implementation is inherently economical in terms of computational complexity.

FIR filters generally require more computations for meeting the same specifications. These observations are true if the input and output sampling rates are the same. But in interpolation, the nature of the problem differs. The output is obtained by filtering the sequence $u_0(n)$ defined by (2.2.7) with a lowpass filter. Consider a zero-phase FIR filter with (2N+1) samples for increasing the sampling rate by a factor L. The choice of an odd number of samples will be explained later.

The output samples for the filter h(n) are

$$y(n) = \sum_{k=n-N}^{n+N} u_0(k)h(n-k) \tag{2.3.1}$$

Since only one out of every L samples of $u_0(n)$ are possibly non-zero, the actual number of computations are reduced by a factor L. This feature is not exploited in conventional IIR filters where the computations must be carried out at the output sampling rate.

Again, in a decimation by a factor M, IIR filters require computation at the input sampling rate as each incoming input sample updates the state variables of the filter. At the output (M-1) out of every M samples are discarded. On the contrary, considerable saving is achieved in an FIR filter by computing only one out of every M samples.

For changing rates by non-integer ratios, say, L/M, we first increase the sampling rate by a factor L and later reduce it by a factor M. The features discussed above can be incorporated in each stage of the process.

2.3(b) Impulse Response Constraints:

One restriction on the Impulse Response arises from the fact that the original samples should be preserved in the output. That is

$$y(mL) = x(m)$$

clearly the requirement is

$$h(o) = 1 \tag{2.3.2}$$

$$h(n) = 0 \quad \text{for } n = \pm L, \pm 2L, \ldots \text{ and } |n| \leq N$$

Another restriction is that the length of the Impulse response should be odd. It can be shown that for even number of samples, a linear phase FIR filter must have a delay of at least one-half sample (25). Such a filter cannot preserve the samples of the original sequence, whereas a filter with an odd number of samples can.

Yet another restriction comes in if we impose the condition that for the computation of each output sample, the number of input samples involved should be the same. For an interpolation by a factor L, the length of the filter response is chosen as $2KL-1$.

2.3(c) Design Specifications:

For interpolation by a factor L, the approximation of the ideal interpolator consists of

(i) specifying a passband within the range $0 \leq w \leq \pi/T$ such that the gain over the band is close to unity. A margin of $\delta_1 \ll 1$ is allowed.

(ii) specifying (L-1) stop-bands centered at $w = (2\pi/T), (4\pi/T), \ldots, 2(L-1)\pi/T$ where the gain is close to zero within the margin $\delta_2 \ll 1$.

For cases where the bandwidth B is close to π/T, the specifications reduce to that for a lowpass filter. There are cases however of narrow band signals where a band stop filter of the same order achieves better results. This is because the filter coefficients can be suitably chosen to cause heavy attenuation only over small bands centered at $2\pi/T$, $4\pi/T, \ldots$.

2.3(d) Efficient Implementation:

For an FIR design, we have not yet discussed schemes which are computationally optimal. Now we see how some features in the interpolation process have been exploited to reduce the number of computations.

Firstly the impulse response for a zero phase FIR filter is symmetric. That is

$$h(n) = h(-n) \tag{2.3.3}$$

This has inherent advantages for an interpolation by a factor 2 as samples equidistant from the point of interpolation can be added first and then multiplied by the filter coefficient. Thus the number of multiplications is halved. This method suggests that if interpolation is carried out in a cascade of filters with 2:1 increase then considerable saving in computations is achieved. Such a procedure has been discussed in (5), (11), and (13). A procedure which incorporates such stages has been presented in (14), where a set of nine half band filters are provided for combination to get the desired sampling rate change.

A multistage procedure to minimise the computations using interpolation ratios which are not necessarily 2:1 has been described in (16). Given the initialsampling rate, the overall interpolation ratio and the filter specifications, the overall sum of the computation rates at each stage is minimised. The flexible parameters for minimisation are the interpolation ratios I_i, $i = 1,2,\ldots,K$ as well as K, the number of stages. Design curves and formulas for the implementation have also been presented in this paper. A similar approach has been used in (17) for minimising the storage rather than the number of computations.

2.4 Digital Interpolation by a Polyphase Network

In (18), a method of manipulating a lowpass filter response to make it amenable to efficient IIR interpolation

has been described. The authors have introduced a structure based on phase-shifters and have referred to it as a poly-phase network. Filtering is carried out by cancellation of the undesired signal components at the output of a summation device connected to a set of phase shifters. The phase shifters are all pass filters and rotate the signal components in such a way that the components to be retained have the same phase at the summation point and add while those to be removed cancel on summation. If an interpolation by a factor L is required, that is, the new sampling period is $T' = T/L$, then the interpolator requires L phase shifters.

Let $H(z)$ be the transfer function a LPF which is obtained by conventional methods to meet the specifications. Suppose

$$H(z) = \frac{A \prod_{k=1}^{K} (z - z_k)}{\prod_{k=1}^{K} (z - p_k)} \qquad (2.4.1)$$

The phase shifters are shown to be a cascade of a delay element and a filter which is a function of z^L. The Polyphase Network transfer function is thus

$$H'(z) = \sum_{n=0}^{L-1} z^{-n} H_n(z^L) \qquad (2.4.2)$$

To determine the coefficients of $H_n(z^L)$ we equate (2.4.1) and (2.4.2). We also note that

$$\frac{1}{z-p_k} = \frac{z^{L-1} + p_k z^{L-2} + \dots + p_k^{L-1}}{z^L - p_k^L} \tag{2.4.3}$$

Using this in (2.4.1) and equating with (2.4.2)

$$\frac{\sum_{i=o}^{KL} a_i z^{-i}}{\prod_{k=1}^{K} (1-p_k^L z^{-L})} = \sum_{n=o}^{L-1} z^{-n} H_n(z^L) \tag{2.4.4}$$

From a simple decomposition of the left hand side, it is seen that

$$H_n(z^L) = [\sum_{k=o}^{K'} a_{kL+n} (z^{-L})^k] / [\prod_{k=1}^{K} (1- p_k^L z^{-L}) \tag{2.4.5}$$

where $K' = K$ for $n = 0$

$= K-1$ for $n > 0$

The fact that H_n is a function of z^{-L} implies that the signal samples coming out of corresponding devices are weighted sums of input samples and previous output samples with N sample spacing. Moreover, from (2.4.5.) we see that the denominators are all identical. This is exploited in the implementation. Figure 2.5 illustrates the implementation for L = 2 and K = 3.

The multiplication rate is $(KL+K+1)f_s/L$ where f_s is the output sampling rate. This shows a considerable

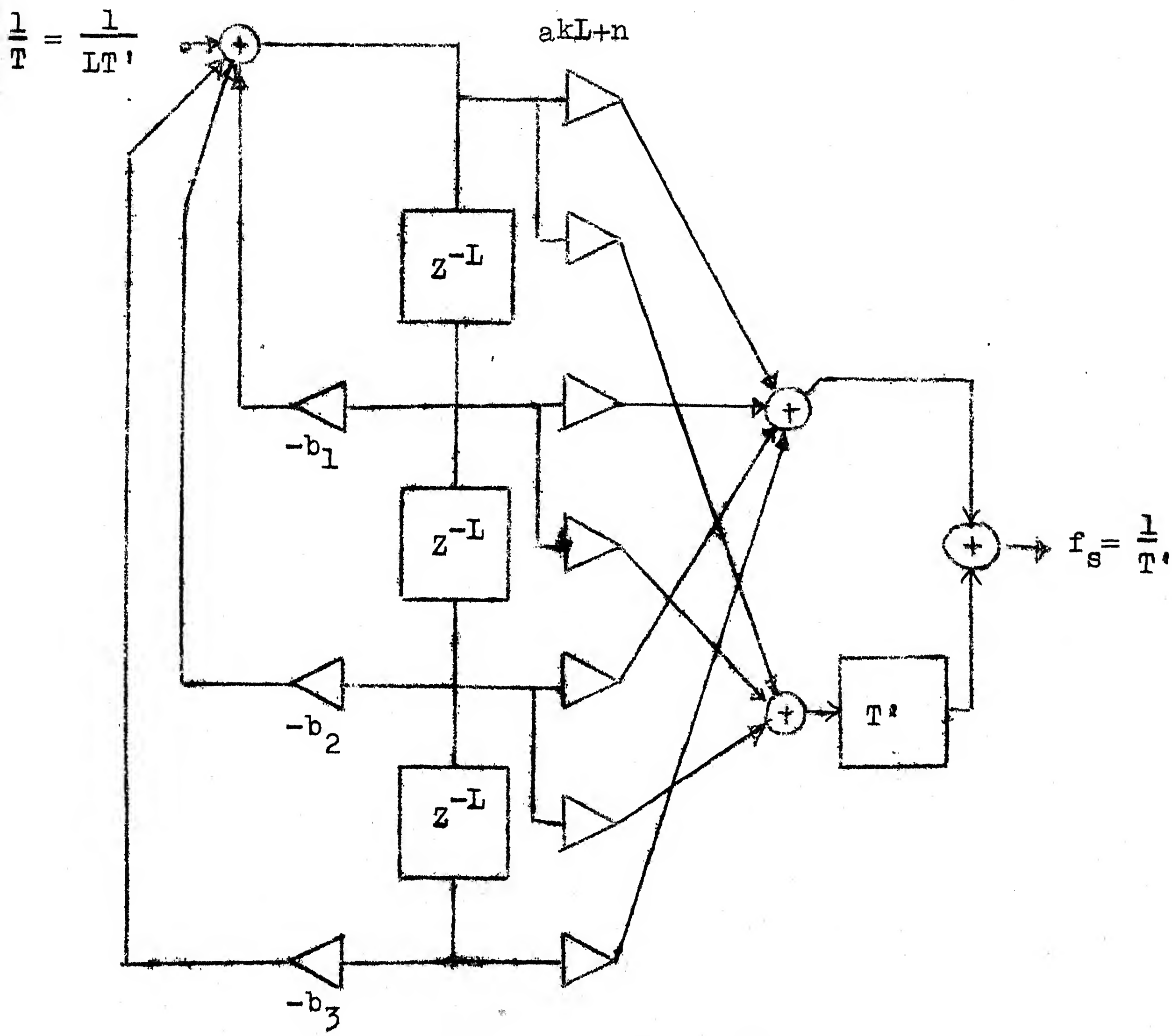


Figure 2.5: Polyphase network for third order lowpass filter and a doubling of sampling rate.

reduction on the 2K f_s computations for the conventional IIR implementation. Comparison with FIR filters should take into consideration the fact that filter coefficients for IIR implementation for given frequency response specifications are fewer. The phase response however, cannot be zero-phase for IIR filters.

2.5 Time Domain Analysis

The problem of interpolation will now be formulated in the time domain using the approach of Oetken et al (19). The analysis has been carried out for noisy samples.

Suppose we seek to increase the sampling rate of a signal by a factor L. The available sequence is x(n), where

$$x(n) = x'(nT)$$

The desired sequence u(n) is

$$u(n) = x'(nT'),\ T' = T/L \qquad (2.5.1)$$

We recall the definition of $u_o(n)$ in (2.2.7) as

$$u_o(n) = x(n/L) = u(n),\ \text{for}\ \ n = mL$$
$$= 0,\ \text{otherwise}$$

The problem is to design the interpolation filter h(n), which, operating on the sequence $u_o(n)$ gives an output y(n) to approximate u(n) according to some error criteria.

We assume that the filter impulse reponse is approximated by $N_T = 2LK-1$ samples where K is an integer, $K \geq 1$. With this we obtain

$$y(n) = \sum_{i=-LK+1}^{LK-1} h(i)\, u_o(n-i) \tag{2.5.2}$$

Since $u_o(n) = 0$ for $n \neq mL$ we get

(i) For $n = mL$ in (2.5.2)

$$y(mL) = \sum_{j=-(K-1)}^{K-1} h(jL)\, u_o((m-j)L) \tag{2.5.3}$$

(ii) For $n = mL+p$; $p = 1,2,\ldots,L-1$

$$y(mL+p) = \sum_{j=-K}^{K-1} h(jL+p)\, u_o((m-j)L) \tag{2.5.4}$$

Since we assumed $h(-KL) = 0$, (2.5.4) holds for $p = 0$. We note that for different values of p, different sample points of h(n) appear in (2.5.4). If we now consider the input sequences to be corrupted by noise, the contribution of noise to the output y(n) can be included in (2.5.4)

$$y(mL+p) = \sum_{j=-K}^{K-1} h(jL+p)[u_o((m-j)L) + w((m-j)L)] \tag{2.5.5}$$

We now define L error sequences

$$\tilde{y}_p(n) = y(n) - u(n) \quad \text{for } n = mL+p$$

$$= 0 \quad , \text{otherwise.}$$

The sequence for total error is

$$\tilde{y}(n) = \sum_{p=0}^{L-1} \tilde{y}_p(n) \tag{2.5.6}$$

A reasonable performance index would be

$$e^2 = \|\tilde{y}\|^2 = \sum_{p=0}^{L-1} \|\tilde{y}_p\|^2 \tag{2.5.7}$$

where $\|\cdot\|$ represents a norm. The norm can suitably be defined for various classes of signals.

Since each error sequence $\tilde{y}_p(n)$ depends on disjoint subsets of h(n), the minimisation of $\|\tilde{y}\|^2$ can be done by minimising each of $\|\tilde{y}_p\|^2$ separately.

Consider x(n) to be a sampled version of a stationary random process with zero-mean and autocorrelation function R(n). For this class of signals the performance measure is chosen to be

$$e_p^2 = E\{\tilde{y}_p^2\}$$

$$= E\{[y(mL+p)-u(mL+p]^2\} \tag{2.5.8}$$

$$e_p^2 = E\{[\sum_{j=k}^{K-1} h(jL+p)(u((m-j)L) + w((m-j)L)) - u(mL+p)]^2\} \tag{2.5.9}$$

Assuming signal and noise to be uncorrelated

$$\frac{\partial e_p^2}{\partial h(iL+p)} = 2[\sum_{j=-K}^{K-1} h(jL+p)[R_u((j-i)L) + R_w((j-i)L)] - R_u(iL+p)] \tag{2.5.10}$$

where $i = -K,\ldots,0,\ldots,K-1$

$$R_u(n) = E\,[u(m)u(m-n)] \tag{2.5.11a}$$

and $$R_w(n) = E[w(m)\,w(m-n)] \tag{2.5.11b}$$

We define $R(n) = R_u(n) + R_w(n)$.

For minimum e_p^2, from (2.5.10) we get

$$\sum_{j=-K}^{K-1} h(jL+p)R((j-i)L) = R_u(iL+p)$$

For convenience we define

$$\underline{R_T} = \begin{bmatrix} R(o) & R(L) & \ldots & R((2K-1)L) \\ R(L) & R(o) & \ldots & \ldots \\ \cdot & \cdot & & \\ \cdot & \cdot & \ldots & \ldots \\ \cdot & \cdot & & \\ R((2K-1)L) & & \ldots & \ldots R(o) \end{bmatrix}$$

$$\underline{H_p} = \begin{bmatrix} h(-KL+p) \\ h(-KL+L+p) \\ \vdots \\ h((K-1)L+p) \end{bmatrix} \quad ; \quad \underline{R_p} = \begin{bmatrix} R_u(-KL+p) \\ R_u(-KL+L+p) \\ \vdots \\ R_u((K-1)L+p) \end{bmatrix} \tag{2.5.12}$$

Then we have

$$\underline{H_p} = \underline{R_T}^{-1}\,\underline{R_p} \tag{2.5.13}$$

The computation of filter coefficients is thus seen to require the inversion of a 2Kx2K matrix and L

multiplications by the vector $\underline{R}_p$. The symmetry of the matrix can be exploited by adding the first and last rows, the second and second last row and so on. This operation allows us to break the system of equations into two sets which require the inversion of two KxK matrices.

For a noiseless signal the choice of h(mL) is the same as in (2.3.2).

The optimum filter thus designed serves as a standard for judging the performance of the other filters. Expressions for error are derived in Appendix I. In the evaluation of the various schemes, the errors will be studied in the context of the optimum interpolator.

CHAPTER 3

DESIGN OF DIGITAL INTERPOLATORS USING AN ESTIMATE OF THE SPECTRUM

In Chapter 2, we had examined approaches to digital interpolation of a signal about which no additional information is available. We also saw that with a knowledge of the autocorrelation function, an optimum filter for minimum mean square error can be designed. In this chapter, we investigate the possibility of improving on the digital interpolation of a signal about which no apriori information is available. The approach suggested is to estimate the spectrum and design an optimum filter using this information. Essentially the idea is to compute the autocorrelation function from an estimate of the spectrum and design an interpolator for minimum mean square error.

The above procedure will be discussed in detail in Section 3.1. In Section 3.2 we shall look at the optimisation procedure in the frequency domain. This helps us to get a better insight into the optimisation. The final results are the same as those obtained in Section 2.5. In Section 3.3, we briefly mention the methods of spectrum estimation with the focus on the Maximum Entropy Method (MEM). Finally we study MEM interpolation by direct operation on the data as suggested in (28).

3.1 Procedure for Obtaining the Approximate Optimum Interpolator

Given the sampled version $x(n)$ of a continuous signal, with a sample spacing T, we again consider the problem of digital interpolation by a factor L. In order to design an optimum interpolator as in 2.5, values of autocorrelation function at a lag $T' = T/L$ and its multiples are required. Clearly by a direct operation on the data, we can, at best, estimate the autocorrelation function at a lag of T and its multiples. To overcome this, we first estimate the spectrum and evaluate the autocorrelation function at the required resolution. As usual, we assume that the signal is bandlimited.

Using the input data we first calculate the spectrum. We use the MEM estimate in view of its recommendation for being non-comittal about data outside the record length and its adaptability to slowly varying statistics. Having obtained an estimate of the spectrum, we can find the autocorrelation function at any desired lag. We can use the FFT algorithm for computing ACF. The next step is to employ the procedure described in Section 2.5 to obtain the optimum interpolator.

How the spectrum estimate helps in bringing about an improvement in performance will be clear from the next section.

3.2 Frequency Domain Interpretation of Optimisation

In Chapter 2, we had seen that interpolation of a signal by a factor L implies that in the frequency domain, we must remove the images of $X'(w)$, the Fourier transform of the continuous signal $x'(t)$, centered at $w = (2\pi/T)$, $(4\pi/T)$, ... $2(L-1)\pi/T$. Refer to Figure 3.1. For the purpose of interpolation, these images can be viewed as noise. The problem is thus analogous to the classical problem of designing a filter for minimum mean square error. In this case the noise and signal occupy different regions of the frequency band.

For optimisation we require the definition of an error measure. This could be achieved by writing the equivalent frequency domain expression for the error in 2.5. We shall obtain the same expression in the following manner.

$x(n)$ is the available sequence and $u_o(n)$ is the sequence obtained by inserting $(L-1)$ zeroes between each pair of adjacent samples of $x(n)$. The autocorrelation function sequence for $x(n)$ and $u_o(n)$ is

$$R(n) = E(x(K)\, x(K+n)) \qquad (3.2.1)$$

$$R_o(n) = E(u_o(K)\, u_o(K+n))$$

Clearly $R_o(n)$ is the sequence obtained by inserting $(L-1)$ zeroes between each adjacent pair of samples of $R(n)$. So

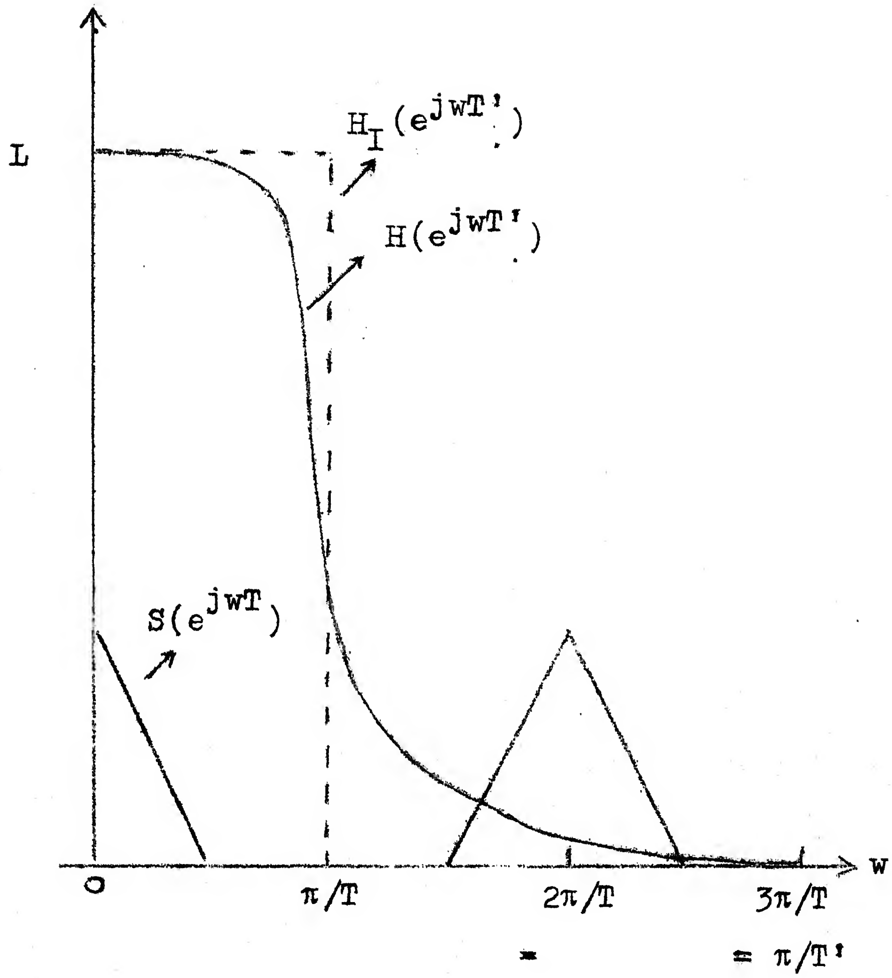


Figure 3.1 Interpolation by a factor $L = 3$.

H_I : Ideal Interpolator

H : Approximation to H_I.

the Fourier transforms of these two sequences are the same.

$$\therefore S_o(e^{jwT'}) = S(e^{jwt}) \qquad (3.2.2)$$

Let $H_I(e^{jwT'})$ be the ideal interpolator and $H(e^{jwT'})$, the approximated interpolator. The error power in the frequency range w and w+dw due to the difference between H_I and H is (but for a constant factor)

$$dp_E = S_o(e^{jwT'})/H_I(e^{jwT'})-H(e^{jwT'})/^2\, dw \qquad (3.2.3)$$

Summing the error power we can express our error measure EM as

$$EM = \int_0^{\pi/T'} S(e^{jwT})/H_I(e^{jwT'})-H(e^{jwT'})/^2\, dw \qquad (3.2.4)$$

We have seen that

$$H_I(e^{jwT'}) = L \quad \text{for } -\pi/T \leq w \leq \pi/T$$

$$= 0 \quad \text{for } \pi/T < |w| \leq \pi/T' \qquad (3.2.5)$$

Therefore,

$$EM = \int_0^{\pi/T} S(e^{jwT})/L - H(e^{jwT'})/^2\, dw$$

$$+\int_{\pi/T}^{\pi/T'} /H(e^{jwT'})/^2\, S(e^{jwT})dw \qquad (3.2.6)$$

We thus see from (3.2.6) that the power spectrum serves as a weighting factor for the error in approximating an ideal

interpolator. The optimum filter would be the one which has a response close to the ideal interpolator over frequency ranges where the signal power is significant. Now, for linear phase,

$$h(n) = h(-n)$$

Using the notations as in Chapter 2,

$$H(e^{jwT'}) = \sum_{i=o}^{K-1} \sum_{p=o}^{L-1} 2\, a_p^i \operatorname{Cos}[(iL+p)wT'] \qquad (3.2.7)$$

where $\quad a_o = \frac{1}{2}h(o)$

and $\quad a_p^i = h(iL+p) \qquad p = 0,1,\ldots,L-1 \qquad (3.2.8)$

$$EM = \int_o^{\pi/LT'} [L-2 \sum_{i=o}^{K-1} \sum_{p=o}^{L-1} a_p^i \operatorname{Cos}[(iL+p)wT']]^2\, S(e^{jwT})dw$$

$$+ \int_{\pi/LT'}^{\pi/T'} [2 \cdot \sum_{i=o}^{K-1} \sum_{p=o}^{L-1} a_p^i \operatorname{Cos}((iL+p)wT')]^2\, S(e^{jwT})dw$$

$$(3.2.9)$$

$\dfrac{\partial\, EM}{\partial a_{p_1}^{i1}} = 0$ for minimum error. From this condition we get

$$2\sum_{i=o}^{K-1} \sum_{p=o}^{L-1} a_p^i \int_o^{\pi/T'} \operatorname{Cos}((iL+p)\omega T') \operatorname{Cos}[(i_1L+p_1)wT']S(e^{jwT})dw$$

$$= L\int_o^{\pi/LT'} \operatorname{Cos}((i_1L+p_1)wT)S(e^{jwT})dw$$

$$\sum_{i=o}^{K-1} \sum_{p=o}^{L-1} a_p^i[\int_o^{\pi/T'} \operatorname{Cos}((\overline{i-i_1}L + \overline{p-p_1})wT')S(e^{jwT})dw \qquad (3.2.10)$$

$$+ \int_o^{\pi/T'} \operatorname{Cos}((\overline{i+i_1}L + \overline{p+p_1})wT')\, S(e^{jwT})dw]$$

$$= L \int_{0}^{\pi/LT'} \mathrm{Cos}((i_1 L+p_1)wT')S(e^{jwT})dw \qquad (3.2.11)$$

$S(e^{jwT})$ is periodic with period $2\pi/T$. Hence the first integral on the left hand side is non-zero for $(p-p_1) = 0$. And the second integral is non-zero for $(p+p_1) = mL$. We therefore write (3.2.11) as

$$\sum_{i=o}^{K-1} a^{i}_{pl} \int_{o}^{\pi/T'} \mathrm{Cos}((i-i_1)wLT')S(e^{jwT})\, dw$$

$$+ \sum_{i=o}^{K-1} a^{i}_{L-pl} \int_{o}^{\pi/T} \mathrm{Cos}((i+i_1+1)wLT')S(e^{jwT})dw$$

$$= \int_{o}^{\pi/T} \mathrm{Cos}((i_1 L+p_1)wT')S(e^{jwT})dw \qquad (3.2.12)$$

where $p_1 \neq 0$.

(3.2.12) is identical to the results obtained in Section 2.5. For p=0, this also yields

$$a^{o}_{o} = \tfrac{1}{2}$$

and $\quad a^{1}_{o} = 0 \quad$ for $i = 1,2,\ldots,K-1 \qquad (3.2.13)$

Let us apply the optimisation to a single tone input with a random phase δ

$$x'(t) = \mathrm{Cos}(w_o t+\delta)$$

$$R(\tau) = \mathrm{Cos}(w_o \tau)$$

$$S(e^{jwT}) = \frac{1}{2T}\,\delta(w-w_o) \quad 0 \leq w \leq \pi/T \qquad (3.2.14)$$

Let L=2, K=2. For p=o, the coefficients a^{i}_{o} are given by (3.2.13). For p=1, we have from (3.2.10)

$$\sum_{i=0}^{1} 2a_1^i \cos((2i+1)w_oT') = 1 \tag{3.2.15}$$

We see that one of the two coefficients, a_1^o or a_1^1 can be arbitrarily fixed. This is apparent as we are required to choose the coefficients so that the gain at the point $w=w_o$ is 2 and is immaterial elsewhere. Choosing $a_1^1 = 0$, we get

$$a_1^o = 1/(2\cos(w_oT'))$$

$$H(e^{jwT'}) = 1+ \frac{1}{\cos(w_oT')} \cos wT' \tag{3.2.16}$$

Clearly at $w=w_o$ the gain is 2 as required. For narrow-band signals it is thus sufficient that the gain be L over a narrow band of frequencies. We can also conceive of an adaptive interpolator for signals with a slowly varying 'spectrum'.

Another point that is clear from the frequency domain interpretation is that for L = 2, the optimisation simplifies considerably. For this case the pass-band and stop band are symmetric and equation (3.2.12) assumes a simple form with $\not{p}_1 = L-p_1$.

We now briefly discuss the methods of estimating the spectrum.

3.3 Spectrum Estimation

Among the conventional methods of spectrum estimation, Blackman-Tukey and Periodogram methods are popular. The Blackman- Tukey approach first estimates the autocorrelation function and then expresses the spectrum as a Fourier transform of this sequence. Autocovariance estimates outside this range are assumed zero. In Periodogram technique, the following estimate is used

$$S'(w) = \frac{1}{N} \left| \sum_{n=1}^{N} x(n)\ e^{-jwnT} \right|^2 \qquad (3.3.1)$$

Here the assumption of a periodic extinsion of data is implicit. These methods suffer from several disadvantages apart from the assumptions in extending the data.

In recent years, the Maximum Entropy Method (MEM) of spectrum estimation has been recommended for its several advantages. We shall not go into the theoretical details of this method. Of the several procedures, we shall use the popular Burg's recursive algorithm (29) for fitting in Autoregressive Model to the input data. This method minimises the error power in passing the input series in the forward and backward direction through the filter. The optimum order of the Autoregressive Model

is determined by Akaike's Final Prediction Error criteria (30). For a detailed study of methods of spectrum estimation with emphasis on MEM estimation, the reader is referred to the thesis (31). This thesis contains useful detailed discussions, programs and extensive bibliography on the subject. The flow chart for the procedure is given in the Appendix II. The details will be clear from the routine SPEMEM in the Appendix.

3.4 Direct MEM Interpolation

This method is suggested in (28). The basic idea is to determine the Fourier Transform of the input signal using the Maximum Entropy Method. From a knowledge of the Transform, the signal samples can be evaluated at any instant of time.

By a classical statistical mechanics argument, the author has related entropy of the sequence to the Fourier Transform by the following equation

$$H = \int_{-B}^{B} \ln[X(w)X^{*}(w)]dw \qquad (3.4.2)$$

This is maximised subject to the constraints

$$\int_{-B}^{B} X(w)\, e^{-jnwT}\, df = x(n) \qquad (3.4.3)$$

where $w = 2\pi f$

and $n = -M, \ldots, 0, \ldots, M$. The solution to the above is shown to be

$$X(w) = [\sum_{n=-M}^{M} b_n^* e^{-jnwT}]^{-1} \qquad (3.4.4)$$

where

$$b_n = \sum_{i=0}^{M-n} c_i \, d_{i+n} \qquad n = 0, \ldots, M$$

$$b_{-n} = \sum_{i=0}^{M-n} c_{i+n} \, d_i \qquad n = 0, \ldots, M$$

and c_i, d_i satisfy

$$\begin{bmatrix} x(0) & x(1) & \ldots & x(M) \\ x(-1) & & & \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ x(-M) & \ldots & \ldots & x(0) \end{bmatrix} \begin{bmatrix} c_0 d_0 \\ c_0 d_1 \\ \cdot \\ \cdot \\ \cdot \\ c_0 d_M \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix}$$

and

$$\begin{bmatrix} x(0) & (x-1) & \ldots & x(-M) \\ x(1) & & & \cdot\cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ x(M) & & & x(0) \end{bmatrix} \begin{bmatrix} d_0 c_0 \\ d_0 c_1 \\ \cdot \\ \cdot \\ \cdot \\ d_0 c_M \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \qquad (3.4.6)$$

Once the coefficients b_n are determined, the Fourier Transform can be computed at any desired resolution using (3.4.4). The interpolation can be carried out using (3.4.3). Clearly in view of the matrix inversion as required in (3.4.6), this method has severe limitations with regard to data length. Also, the number of computations per interpolated sample is extremely high compared to conventional interpolation.

A routine MEMINT for performing interpolation using this technique is included in the Appendix.

CHAPTER 4

RESULTS OF STUDIES USING A REPRESENTATIVE SET OF SIGNALS

This chapter presents the results of application of the interpolation techniques discussed in Chapters 2 and 3 to a representative set of signals. In Section 4.1 we give a description of the various signals employed for studying the performance of the interpolation techniques and the filters used for interpolation. In Section 4.2 we list the results of studies on deterministic signals and in Section 4.3 on random signals.

4.1 Description of the Signals and Filters used for the Study

For a meaningful comparative study, it is necessary that the conditions and restrictions on the various digital interpolators be identical. In this study the initial sampling frequency is chosen as 1 Hz. The signal is considered bandlimited to 0.5 Hz. The factors by which the sampling rate is sought to be increased are chosen to be L=2 and L=4. Four neighbouring points are used for interpolation of each sample. Thus the FIR filters used have an impulse response of length 7 and 15 for the cases L=2 and L=4 respectively. The input record length was chosen as 128.

4.1(a) Signals used:

Only real input series will be considered. The performance will be evaluated on the basis of the mean square error. For obtaining the error, the samples of the continuous signal at the points of interpolation should be available. The approach in the study has been to first obtain a series at the desired final sampling rate. This series is then decimated to get the input data for interpolation. The error sequence is thus the difference between the interpolated sequence and the undecimated input sequence.

Both deterministic and random signals have been used in the study. Single tone inputs, sums of sinusoids and Sinc functions are the deterministic signals used. The random signals were generated by processing a White noise series, generated by the method of multiplicative noise (routine WGNOIZ). The white noise series was passed through a single first order lowpass filter or a cascade of these filters. Abother series was obtained by summing up 24 consecutive signal samples in the white series. This series has a triangular autocorrelation function and is useful for its analytical tractability.

$$x(k) = \sum_{i=1}^{24} n(k+i-24) \qquad (4.1.1)$$

where n(k) are white noise samples.

The autocorrelation function is

$$R(n) = E(x(k)x(k+n)) \qquad (4.1.2)$$

$$R(n) = \sigma^2(24 - n) \quad \text{for} \quad n \leq 24$$
$$= 0 \quad \text{for } n > 24 \qquad (4.1.3)$$

where $\quad E(n(K)n(K+n)) = \sigma^2\delta_{no} \qquad (4.1.4)$

4.1(b) Filters used:

The notations in the brackets will be used to refer to the filters.

(i) Truncated Sinc function (TSINC): In this case the finite duration impulse response is a truncated sinc function extending over 4 seconds, i.e. four sampling periods of the input signal.

(ii) Cubic Spline Response (CSPLIN): A finite duration cubic spline curve used for approximating a sinc function is mentioned in (32). The conditions used in the approximation are a duration of four sample periods, slope and value continuity, symmetry about the time origin, and a unit area under the curve. The resulting cubic convolution polynomial is

$$h(t) = \begin{cases} 1-2|t|^2 + |t|^3, & |t| < 1 \\ 4-8|t| + 5|t|^2 - |t|^3, & 1 \leq |t| < 2 \\ 0, & |t| \geq 2 \end{cases}$$

(4.1.5)

(iii) Lagrange Interpolator (LGRNJ): A four point Lagrange interpolator has been used in the study. The impulse response samples for cases (i), (ii) and (iii) are listed in Table I. The coefficients for each interpolated sample are given row-wise. The impulse response and the corresponding frequency response are shown in Figure 4.1.

(iv) Optimum Interpolators (OPTIM): We have used four point optimum interpolators derived from either a knowledge of the autocorrelation function (OPT I) or by the use of MEM spectrum estimate as discussed in Chapter 3. (OPTMEM).

(v) Polyphase Network (PPNET): IIR interpolators using phase-shifters were discussed in Section 2.4. This technique has been employed for doubling the sampling rate with a third order Butterworth filter. The details of the filter coefficients are provided in Table II. See Figure 2.5 and Section 2.4 for details.

(vi) Direct MEM Interpolation (MEMINT): The method outlined in Section 3.4, has been applied for interpolating some signals. In this case the record length was small.

4.2 Results for Deterministic Signals

The criteria used in evaluating the performance of interpolators is

TABLE I. IMPULSE RESPONSE COEFFICIENTS

(1) TRUNCATED SINC FUNCTION INTERPOLATOR

0.00000000	0.00000000	1.00000000	0.00000000
-0.12861666	0.30010548	0.90031632	-0.18006324
-0.21220660	0.63661980	0.63661980	-0.21220660
-0.18006324	0.90031632	0.30010548	-0.12861666

(2) CUBIC SPLINE INTERPOLATOR

0.00000000	0.00000000	1.00000000	0.00000000
-0.04687500	0.29687500	0.89062500	-0.14062500
-0.12500000	0.62500000	0.62500000	-0.12500000
-0.14062500	0.89062500	0.29687500	-0.46875000

(3) LAGRANGE INTERPOLATOR

0.00000000	0.00000000	1.00000000	0.00000000
-0.03906250	0.27343750	0.82031250	-0.05468750
-0.06250000	0.56250000	0.56250000	-0.06250000
-0.05468750	0.82031250	0.27343750	-0.03906250

TABLE II. POLYPHASE NETWORK

POLES IN THE Z-PLANE

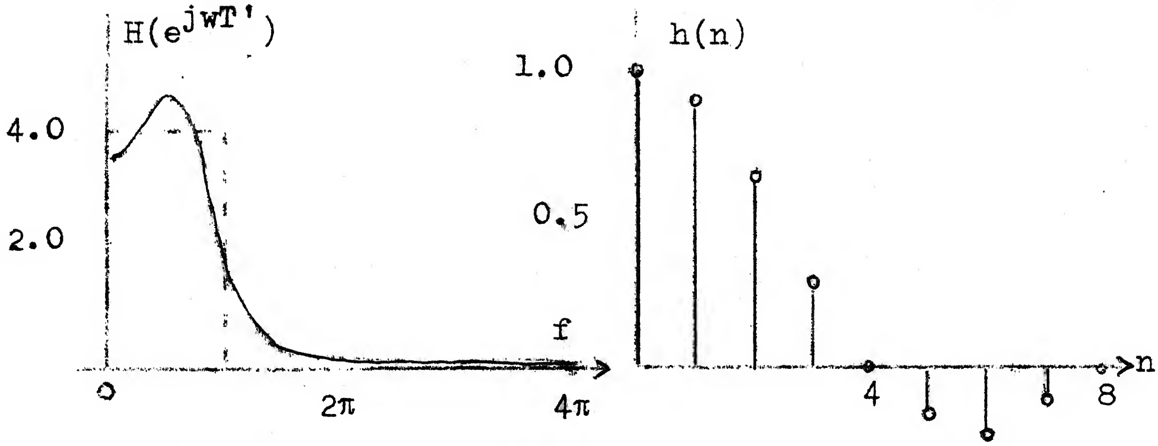
P(1)=0.26794919+J0.00000000
P(2)=0.34891526-J0.52337289
P(3)=0.34891526+J0.52337289

PHASE SHIFTER COEFFICIENTS

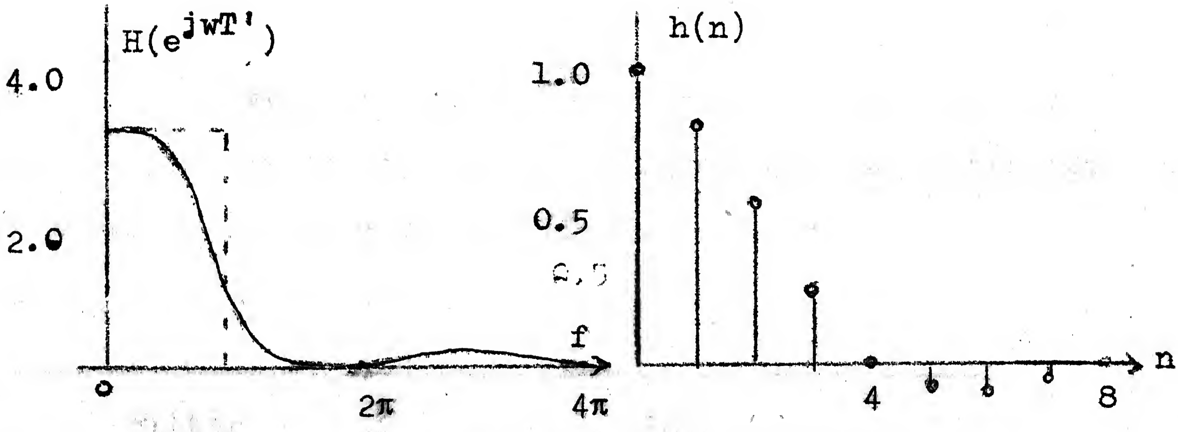
A(1)=1.00000000 A(2)=3.96577968 A(3)=6.47998320
A(4)=5.75128856 A(5)=3.03176328 A(6)=0.90069530
A(7)=0.10601705

DENOMINATOR COEFFICIENTS

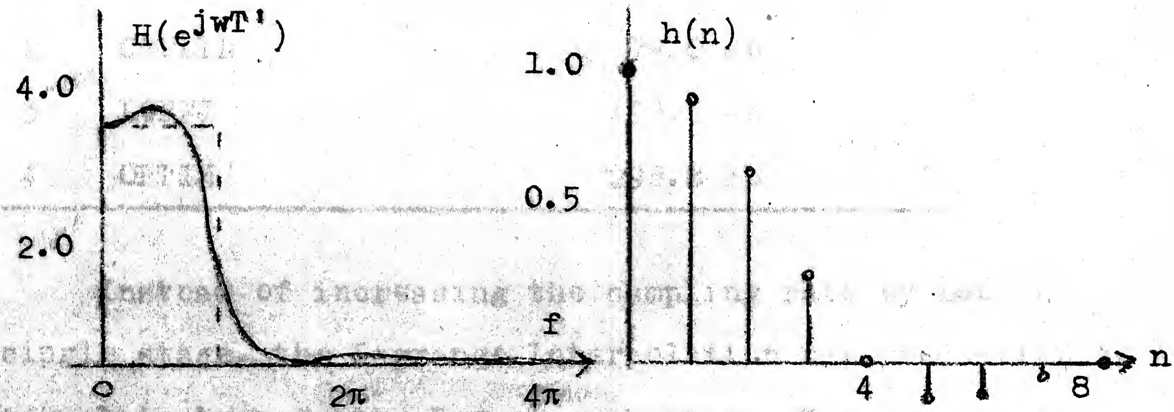
B(1)=0.23255787 B(2)=0.13469597 B(3)=-0.01123962



(a) Truncated Sinc.



(b) Lagrange.



(c) Cubic Spline.

Figure 4.1: Frequency Response $H(e^{jwT'})$ and impulse response $h(n)$ of Interpolators.

$$(S/N) = \frac{\sum u^2(n)}{\sum [y(n)-u(n)]} \qquad (4.2.1)$$

where $y(n)$ is the interpolated series and $u(n)$ is the undecimated input sequence. The summation is over all interpolated samples.

4.2.1 Single tone input:

Listed in Table III are results for a sampled sinusoid of frequency 0.025 Hz. Using the approach of Section 3.2, the theoretically designed optimum filter for the known frequency was used (L=4).

TABLE III: Single tone input (f=0.025 Hz)

	Filter	(S/N)
1	TSINC	39.4 db
2	CSPLIN	79.0 db
3	LGRNJ	199.7 db
4	OPTIM	298.8 db

Instead of increasing the sampling rate by L=4 in a single stage, the Lagrange Interpolation was used twice to interpolate by a factor L=2 at each stage. The results are given in Table IV.

Table IV: Single tone input
(f = 0.025 Hz)

Filter	S/N
1. Single stage LGRNJ	199.7 db
2. Two stage LGRNJ	201.8 db

Interpolation of a single tone input by a factor L=2 was carried out by using a polyphase network. For an input tone frequency of 0.03125 Hz, a phase lag of 0.025935 radians was observed. The error was computed after eliminating the phase lag. The results is given in Table V.

Table V: Single tone input
(f = 0.03125 Hz)

Filter	(S/N)
1. PPNET	150.5 db
2. LGRNJ	132.3 db

Direct MEM Interpolation is not applicable to this case. The solution matrix for a periodic signal is singular.

4.2.2 Sum of Sinusoids:

First we consider a sum of two sinusoids at $f_1 = 0.125$ Hz and $f_2 = 0.333$ Hz with equal amplitudes.

If the frequencies are known, an approach similar to that given in 3.2 can be used in designing the interpolator for exact interpolation. In the present study, the spectrum was estimated by using the MEM approach with a maximum AR model order 10. The spectrum estimate is shown in Figure 4.2.

The theoretically computed impulse response coefficients for exact interpolation for a factor L=2 and those obtained by MEM spectrum estimate are given in Table VI.

Table VI: Filter Coefficients for Intermediate Points with L =2.

Filter	$a_1^1 = a_1^{-1}$	$a_1^2 = a_1^{-2}$
1. Theoretical Optimum	0.6199144	-0.1920603
2. OPTMEM	0.6208004	-0.1921782

A comparative study of the performance with a Lagrange interpolator can be made from the following table.

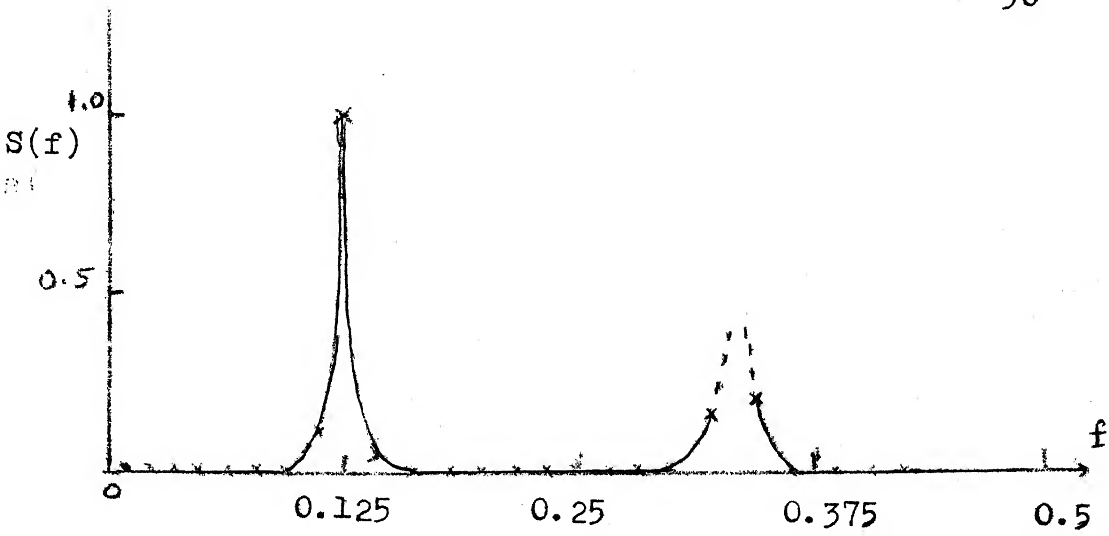


Figure 4.2: MEM Spectrum for a sum of Sinusoids.

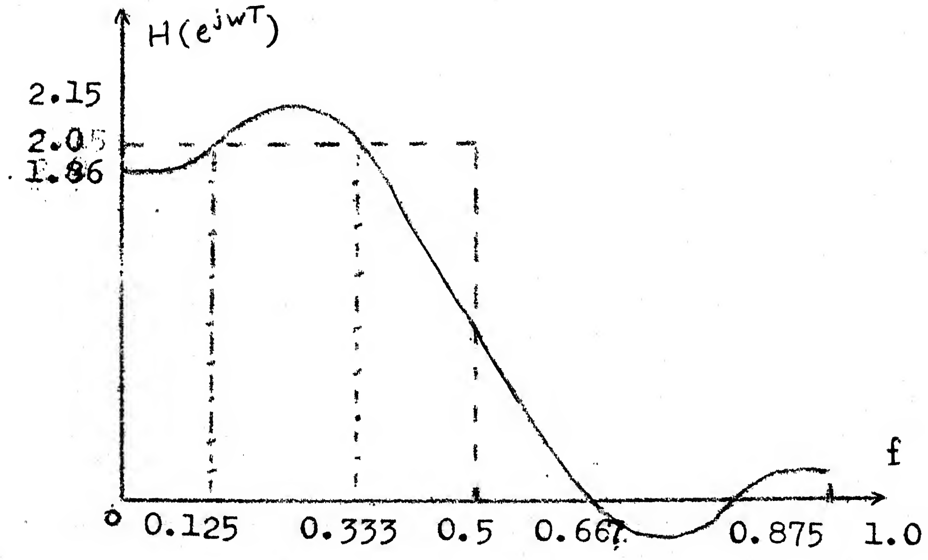


Figure 4.3: Frequency response of the optimum filter for a sum of sinusoids obtained by an MEM spectrum estimate. (f_1 = 0.125 Hz and f_2 = 0.333 Hz).

Table VII: Sum of two sinusoids (L=2)
(f_1 = 0.125 Hz and f_2 = 0.333 Hz)

Filter	(S/N)
1. OPTMEM	109.6 db
2. LGRNJ	38.3 db

The filter response is shown in Figure 4.3. The filter gain at f_1 and f_2 is 2.0000013 and 2.0051568 respectively against the required value of 2.0.

Another study was made using a sum of five sinusoids closely spaced in frequency to get a narrowband signal.

$$x(n) = \sum_{i=-2}^{2} \mathrm{Sin}(2\pi n(0.145 + 0.01\, i)) \qquad (4.2.2)$$

The results are listed in Table VIII.

TableVIII Results for sum of 5 sinusoids
(Eqn. 4.2.2)

	Filter	(S/N)
1	TSINC	67.0 db
2	CSPLIN	61.4 db
3	LGRNJ	90.7 db
4	OPTMEM	145.91 db

4.2(c) Sinc Function:

The sampled sinc function serves as a convenient deterministic function for studying direct MEM interpolation. A 9-point input was chosen to cover the central lobe and the first sidelobe on either side. The results are listed below.

Table IX: Interpolation of a sampled Sinc function (L=2).

	Filter	(S/N)
1	TSINC	48.4 db
2	CSPLIN	64.1 db
3	LGRNJ	112.9 db
4	MEMINT	132.0 db

4.3 Random Signal Interpolation

We first consider the signal described by eqn.(4.1.1) for which the autocorrelation function is triangular. The optimum interpolator for this case is the linear interpolator. The results for this case are listed in Table X.

Table X: Interpolation for Signal with a Triangular Autocorrelation function.

Filter	(S/N)
TSINC	15.66 db
CSPLIN	17.84 db
LGRNJ	18.99 db
LINEAR INTERPOLATOR	19.18 db
KOPTMEM	17.95 db

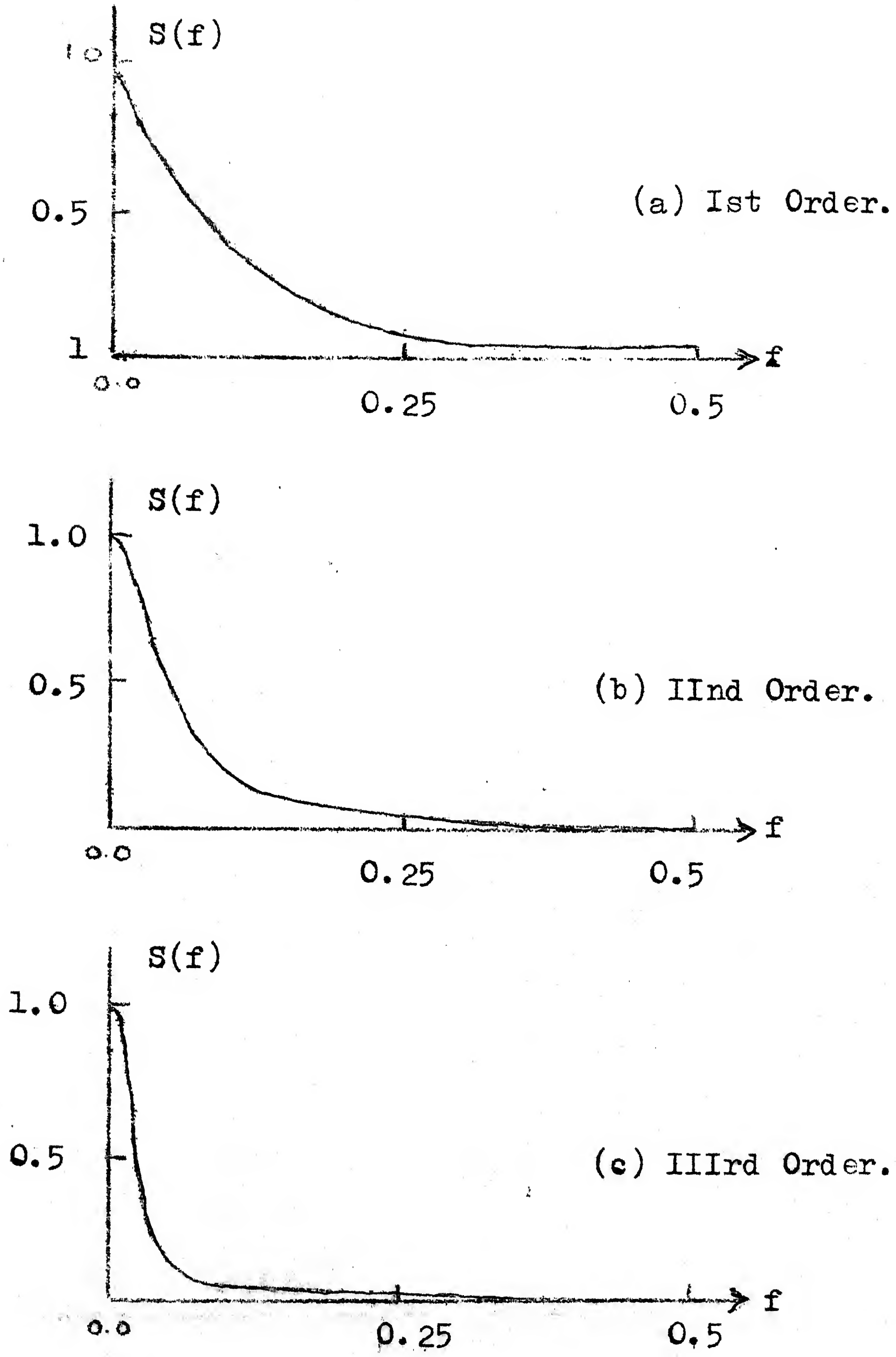


Figure 4.4: Spectrum for A = 0.9, B = 0.1.

Next we consider a white noise series passed through a first order filter of the form

$$x(k) = A\, x(k-1) + Bn(k) \qquad (4.3.1)$$

Second and third order filters were realised as cascades of identical first order filters. The normalised spectrum, with $A = 0.9$ and $B = 0.1$, for first, second and third order filters is shown in Figure 4.4.

We shall now list the results for white noise series, both uniformly distributed and Gaussian, passed through second and third order filters. We first consider white noise passed through the following filters. (Table XI, XII).

Table XI: Second Order Filter
($A = 0.95$, $B = 0.05$).

	Filter	(S/N)
1.	TSINC	41.37 db
2.	CSPLIN	65.75 db
3.	LGRNJ	69.92 db
4.	OPTMEM‡	70.27 db
5.	OPTMEM‡‡	(69.68) db

OPTMEM‡‡ : This is the optimum interpolator derived from the MEM estimate computed at a low frequency resolution (1/4 of that used in 4).

The results which will be listed were obtained for Gaussian white noise passed through second and third order filters. (Table XIII - XVI).

Table XII: Third Order Filter
(A = 0.9, B = 0.1)

	Filter	(S/N)
1.	TSINC	(39.70 db
2.	CSPLIN	67.36 db
3.	LGRNJ	84.19 db
4.	OPTMEM	85.34 db

Table XIII: (A = 0.8, B = 0.15)

Filter	Second order (S/N)	Third order (S/N)
1. TSINC	29.63 db	36.81 db
2. CSPLIN	31.31 db	51.57 db
3. LGRNJ	32.33 db	58.92 db
4. OPTMEM	32.20 db	59.46 db

Table XIV: (A = 0.85, B = 0.1)

Filter	Second order (S/N)	Third order (S/N)
1. TSINC	36.55 db	42.50 db
2. CSPLIN	43.68 db	66.00 db
3. LGRNJ	44.32 db	74.84 db
4. OPTMEM	44.65 db	75.60 db

In the following table, we list results for the case where the sample realisations used for computing the spectrum and for interpolation are different. That is, the white noise realisations that were passed through the third order filter were different.

Table XV: (A = 0.9, B = 0.1)

	Filter	Second order (S/N)	Third order (S/N)
1.	TSINC	41.51 db	41.12 db
2.	CSPLIN	62.87 db	77.82 db
3.	LGRNJ	65.90 db	86.27 db
4.	OPTMEM	66.36 db	86.51 db

Two seven point series were used for direct MEM interpolation. In one case the input samples were preserved in the output. In the second, the output was not consistent with the input. For the first case, the result was

Table XVI: Second Order Filter

(A = 0.95, B = 0.05)

Filter	(S/N)
MEMINT	55.28 db
LGRNJ	59.24 db

CHAPTER 5

CONCLUSIONS

In this chapter we analyse the results presented in the last chapter. The results for the deterministic signals will be discussed only to the extent that they shed light on the processing of random signals. On the basis of the performance, some comments and recommendations regarding the various interpolators are made.

In our studies, only a four point interpolation has been used. This condition imposes a restriction on the approximation to the lowpass filter that can be obtained. In order to make a comparative study, we choose the Lagrange Interpolator as a standard for evaluation. The Lagrange interpolator is an approximation to the ideal interpolator with a relatively flat response at the origin. In the absence of any knowledge about the signal, it would seem that the Lagrange interpolator would be the best solution for interpolation. This observation is confirmed by studies on signals with frequency content very low compared to half the sampling frequency.

In studying the interpolation for a single tone input, the optimum interpolator was derived assuming a knowledge of the frequency. Exact interpolation is possible here as the optimum filter is essentially a solution to a

set of equations with two unknowns, the phase and the amplitude. A comparison of the other filters with this optimum is not meaningful. But the results show that the Lagrange interpolator performs well at the frequency $f = 0.00625$ Hz. (Table III). The TSINC and CSPLIN performance is far inferior. This is to be expected in view of their frequency response close to the origin.

In the interpolation of a sum of two sinusoids the frequencies chosen were high (1/4 and 2/3 of half the sampling frequency). Here instead of assuming any knowledge of the input, we used the suggested method of estimating the MEM spectrum and designing the interpolator. The filter coefficients computed are quite close to the theoretical optimum (Table VI). The results in Table VII show the wide difference in the performance between OPTMEM and LGRNJ interpolators. Thus for signals with a large high frequency content, the spectrum estimation enables us to design interpolators for far better performance. The implications for interpolating a periodic signal are also clear. A gain of L at the various harmonics is adequate for exact interpolation.

Next we look at the results in Table VIII for a sum of five closely spaced sinusoids. Here too the OPTMEM interpolator is far superior to LGRNJ. The gains of these two interpolators at the mid-frequency of the

sinusoids were 4.000839 and 3.963334. Thus if the estimate indicates a narrow band of frequencies, the interpolator can be designed with considerable saving in filter length to meet the ideal interpolator requirements. If we desire to increase the sampling rate of a multiplexed signal, say of five composite signals, then the optimum interpolator would be required to approximate the ideal interpolator in the region around the five carrier frequencies with attenuation at the unwanted images.

From Table IV we see that the two stage Lagrange interpolator performs slightly better than a single stage filter. The frequency response of the cascaded system at the input frequency is closer to the ideal interpolator than the single stage system. This shows that a cascaded system can be judiciously designed so that the final gain in the frequency band of interest is close to L, the interpolation ratio. This achieves a considerable reduction in computations.

Table V shows that polyphase implementation performs well but for phase delay. In cases where phase distortion is not important, e.g. speech signals, polyphase implementation can be used. An advantage in computation results if the interpolation ratio is high.

In studying the results for random signals, we shall first check how close to the theoretically expected

minimum error performance are the results actually obtained. Using the method of error computation in Appendix I, the (S/N) ratio for the optimum, in this case, a linear interpolator was found to be 19.20. The value obtained from the implementation is 19.18. For this signal the OPTMEM performance is inferior to the Lagrange interpolator. This is because the signal spectrum has considerable high frequency content and this shows up as aliasing in the spectrum estimate. The optimum interpolator design thus uses an incorrect weighting in computing the filter coefficients.

Now we consider the signals where white noise is passed through a second and third order filters. In all cases except one, the OPTMEM performance is superior to Lagrange interpolation. This has to be seen in the light of the fact that that spectrum is significant only in regions close to the origin. In this region, the Lagrange Interpolation approximates the ideal interpolator very closely. The exception is in the case of the second order filter (Table XIII), where there is some aliasing in view of the choice of filter coefficients.

It is important that the spectrum be evaluated at a sufficiently high resolution as a computation using a low resolution spectrum may introduce errors in filter coefficients. This may be seen from Table XI.

If the spectra have a considerable high-frequency content then the performance could be improved greatly by using the spectral estimate. If the signal statistics are slowly varying, the spectrum can be evaluated periodically and interpolation coefficients updated accordingly.

So far as MEMINT is considered, its application requires that to get a close estimate of the spectrum all frequencies should show up in the record. This would demand a large length of the record. On the other hand, the solution matrix becomes increasingly difficult to solve with larger lengths. For symmetric data the performance is found to be consistent but in other cases some inconsistencies showed up. From the point of view of computational simplicity, this method is hardly competitive.

We find that the Truncated Sinc and cubic spline interpolators invariably showed a poor performance. Another result that emerges is the consistently good performance of Lagrange interpolator. But the OPTMEM interpolator performance shows that an improved design is possible with an estimate of the spectrum. This method is particularly useful for stationary random signals with large record length. This method can be advantageously used for narrow band signals.

APPENDIX I

EXPRESSION FOR INTERPOLATION ERROR FOR STATIONARY SERIES

The error measure is given by (see eqn.2.5.9)

$$e_p^2 = E\left\{\left[\sum_{j=-K}^{K-1} h(jL+p)u((m-j)L) + w((m-j)L) - u(mL+p)\right]^2\right\} \tag{A.1.1}$$

Assuming that the noise is uncorrelated with the signal

$$e_p^2 = \sum_{i=-K}^{K-1} \sum_{j=-K}^{K-1} h(jL+p)h(iL+p)[R_u((i-j)L) + R_w((j-i)L)] + R_u(o) - 2\left[\sum_{j=-K}^{K+1} h(jL+p)R_u(jL+p)\right] \tag{A.1.2}$$

For minimum error

$$\frac{\partial e_p^2}{\partial h(iL+p)} = 0$$

This gives

$$\sum_{j=-K}^{K-1} h(jL+p)R((j-i)L) = R_u(iL+p) \tag{A.1.3}$$

where

$$R(n) = R_u(n) + R_w(n) \tag{A.1.4}$$

Therefore,

$$e_p^2 \min = R_u(o) - \sum_{j=-K}^{K-1} h(jL+p)R_u(jL+p)$$

Let $\alpha_{ij} = R_u((j-i)L)/R_u(0)$

$\beta_{ij} = R_w((j-i)L)/R_u(0)$

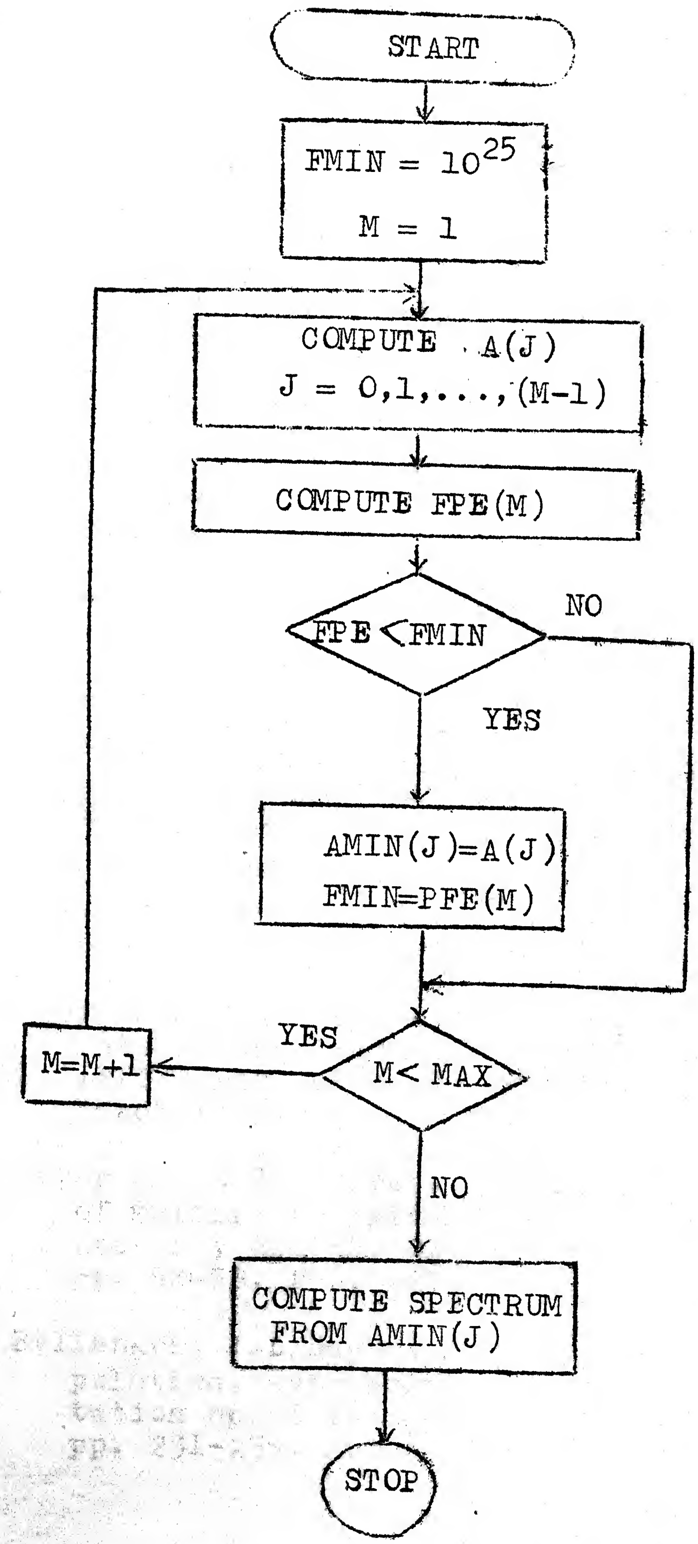
and $\gamma_{ip} = R_u(iL+p)/R_u(0)$

$$S/N = \frac{E[u^2(n)]}{\frac{1}{L}\sum_{p=o}^{L-1} e_p^2}$$

$$= \left\{\frac{1}{L}\sum_{p=o}^{L-1}\left[1 + \sum_{i=-K}^{K-1}\sum_{j=-K}^{K-1} h(iL+p)h(jL+p)[\alpha_{ij}+\beta_{ij}] - 2\sum_{j=-K}^{K-1} h(jL+p)\gamma_{ip}\right]\right\}^{-1}$$

$$(S/N)_{min} = \left[1 - \frac{1}{L}\sum_{p=o}^{L-1}\sum_{j=-K}^{K-1} h_{min}(jL+p)\gamma_{ip}\right]^{-1}$$

APPENDIX II



Flow chart for computing the MEM Spectrum.

REFERENCES

1. R.W. Schefer and L.R. Rabiner, 'A Digital Signal Processing approach to Interpolation', Proc. IEEE, vol.61, pp. 692-702, June 1973.

2. J.L. Flanagan, Speech Analysis, Synthesis and Perception, New York: Springer Verlag, 1965.

3. R.W. Shafer and L.R. Rabiner, 'Design and simulation of a speech analysis-synthesis system based on short time Fourier analysis', IEEE Trans. Audio Electroacoust., vol.AU21, pp. 165-174, June 1973.

4. D.J.Goodman, 'The application of delta modulation to analog-to-PCM encoding', BSTJ, vol.48, pp.321-344, Feb. 1969.

5. D.J. Goodman, 'Digital filters for code format conversion', Elect.Letters, Vol.11, pp.89-90, Feb.20, 1975.

6. S.L. Frecney, R.B.Kierbutz, K.V.Mina and S.K.Tewksbury, 'Systems analysis of a TDM-FDM Translator/Digital A-type channel bank', IEEE Trans. Comm.Tech., Vol. COM-19, No.6, pp. 1050-1060, Dec. 1971.

7. A.W.Crooke and J.W.Craig, 'Digital filters for sample rate reduction', IEEE Trans. Audio Electroacoust., vol. AU-20, pp. 223-225, August 1972.

9. R.A. Meyer and C.S. Burrus, 'A unified analysis of multirate and periodically time-varying digital filters', IEEE Trans. Circuits Syst., CAS-22, pp. 162-168, March 1975.

10. R.A. Meyer and C.S. Burrus, 'Design and Implementation of multirate digital filters', IEEE Trans. Acoust., Speech, Signal Processing, vol.ASSP-24, pp. 53-58, Feb. 1976.

11. M.G. Bellanger, J.L.Daguet and G.P.Lepagnol, 'Interpolation, extrapolation and reduction of computation speed in digital filters', ASSP-22, pp. pp. 231-235, August 1974.

12. L.R. Rabiner and R.E.Crochiere, 'A novel implementation for narrow band FIR digital filters', ASSP-23, pp. 459-464, Oct. 1975.

13. D.W. Rorabacher, 'Efficient FIR filter design for sample rate reduction or interpolation', Proc. IEEE, Int.Sym.on Circuits and Systems, Apr. 21-23,1975, pp. 396-399.

14. D.J.Goodman and M.J.Carey, 'Nine digital filters for decimation and interpolation', ASSP-25, pp. 121-126, April 1977.

15. R.R. Shively, 'On multistage FIR filters with decimation', ASSP-23, pp. 253-257, Aug. 1975.

16. R.E. Crochiere and L.R.Rabiner, 'Optimum FIR digital filter implementations fordecimation, interpolation and narrow band filtering', ASSP-23, pp. 444-458, Oct. 1975.

17. R.E. Crochiere and L.R. Rabiner, 'Further considerations in the Design of Decimators and Interpolators', ASSP-24, pp.296-311, Aug.1976.

18. M.G. Bellanger, G. Bonnerot and M. Coudreuse, 'Digital filtering by polyphase network: Applications to sample-rate alteration and filter Banks', ASSP-24, pp. 109-115, April 1976.

19. G. Oetken, T.W. Parks and H.W. Schussler, 'New results in the Design of digital interpolators', ASSP-23, pp. 301-309, June 1975.

20. P. Castellino and C.Scagliola, 'On optimum FIR interpolators for sampled signals', CSELT Rapporti tecnici, No.3, pp.3-9, Sept.1975.

21. D. Malah and M.Botbol, 'Time-limited signal interpolation by means of the FFT', The ninth Convention of Electrical and Electronics Engineers in Israel, April 1975.

22. W.D.Stanley, 'Digital Signal Processing, Reston Va.: Reston Publishing Company, 1975.

23. L.R. Rabiner, 'Techniques for designing FIR digital filters', IEEE Trans. Commun. Technol., vol.Com-19, pp. 183-195, April 1971.

24. L.R. Rabiner, 'The design of FIR digital filters using linear programming techniques', BSTJ, vol.51, pp. 1177-1198, July-Aug.1972.

25. L.R. Rabiner and R.W.Schafer', Recursive and non-recursive realizations of digital filters designed by frequency sampling techniques', IEEE Trans. Audio Electroacoust., AU-19, pp. 200-207, Sept. 71.

26. C.E. Froberg, Introduction to Numerical Analysis, Addison Wesley Publishing Company, 1970, pp. 164-174.

27. R.M. Lerner, 'Representation of Signals', Lectures on Communication System Theory, ed. E.J. Baghodady, New York: McGraw-Hill,1961, pp.205-210.

28. W.I.Newman, 'Extensions to the maximum entropy method', National Astronomy and Ionosphere Centre and Cornell University Publication, Jan. 1974.

29. J.P. Burg, 'Maximum entropy spectral analysis', Paper presented at the 37th Annual International Meeting, Society of Explor. Geophys., Oklahoma, 1967.

30. H.Akaike, 'Power Spectrum estimation through autoregressive model fitting', Am.Inst.Stat.Math., vol.21, 1969, pp. 243-247.

31. B.V.K.Vijayakumar, 'Power spectrum estimation and maximum entropy method', M.Tech. Thesis, Department of Electrical Engineering, Indian Institute of Technology, Kanpur.

32. R. Bernstein, 'Digital image processing of earth observation sensor data', IBM Journal of Research and Development, Vol.20, No.1, PP 40-57, Jan. 1976.

THE FOLLOWING ROUTINES ARE LISTED

1. INTERP: Interpolation by a factor L using M neighbouring samples
2. SPEMEM: Spectrum Estimation by Maximum Entropy Method (MEM)
3. MEMINT: Direct Interpolation
4. OPTINT: Optimum Filter Coefficients for L=2 and M = 4.
5. WGNOIZ: White Noise Series Generation - Uniformly Distributed or Gaussian.

PROGRAM

```
SUBROUTINE INTERP(X,NP,H,MP,L,Y)
THIS ROUTINE PERFORMS DIGITAL INTERPOLATION BY AN INTEGER FACTOR L
 X IS THE INPUT SERIES
 NP IS THE NUMBER OF INPUT DATA POINTS
 L IS THE FACTOR BY WHICH THE SAMPLING RATE IS TO INCREASED
 H(I,J) ARE THE IMPULSE RESPONSE COEFFICIENTS WHERE I RANGES FROM
    1,2,....,L AND J RANGES FROM 1,2,....,MP. FOR I=1 THE ORIGINAL
    SAMPLES IN THE INPUT SERIES ARE RETAINED
 MP IS THE NUMBER OF INPUT POINTS TO BE USED FOR INTERPOLATION
    AT EACH POINT. MP IS EVEN. FOR I=2,3,....,MP (MP/2) POINTS ON
    EITHER SIDE OF THE INTERPOLATION INSTANT ARE USED
 Y IS THE OUTPUT SERIES
DIMENSION Y(L*NP),X(NP),W(NP+MP-2),H(L,MP)
DIMENSION Y(1000),X(200),W(250),H(10,10)
MPP=MP/2-1
DO 11 I=1,MPP
II=NP+MPP+I
W(I)=0.0
W(II)=0.0
DO 21 I=1,NP
IMP=I+MPP
W(IMP)=X(I)
NP1=NP-1
DO 31 K=1,NP1
DO 31 I=1,L
M=L*(K-1)+I
Y(M)=0.0
DO 31 J=1,MP
KJ=K+J-1
JJ=MP-J+1
Y(M)=Y(M)+H(I,JJ)*W(KJ)
RETURN
END
```

PROGRAM

```
SUBROUTINE SPEMEM(X,NDATA,MAX,FINTVL,TINTVL,SPECTR,MIN,FMIN,NFREQ)
SPECTRUM EVALUATION BY MAXIMUM ENTROPY METHOD. BURG'S ALGORITHM
COUPLED WITH AKAIKE'S FINAL PREDICTION ERROR CRITERIA IS USED
IN THE FIRST STAGE THE AUTOREGRESSIVE MODEL COEFFICIENTS AND
THE FINAL PREDICTION ERROR FOR EACH ORDER OF THE MODEL,M=1,.. .MAX
ARE COMPUTED. THE A R MODEL COEFFICIENTS FOR THE MINIMUM F P E
ARE STORED IN AMIN(J). THESE COEFFICIENTS ARE THEN USED IN
COMPUTING THE SPECTRUM. IF THE AUTOCORRELATION FUNCTION IS TO
BE EVALUATED USING A RADIX 2 F F T  ROUTINE FINTVL SHOULD BE SO
CHOSEN THAT NFREQ IS OF THE FORM 2**K
 X  IS THE INPUT SERIES
 NDATA IS THE NUMBER OF DATA POINTS
 MAX    IS THE MAXIMUM ORDER OF THE A R MODEL TO BE FITTED
 FINTVL IS THE FREQ. INTERVAL AT WHICH THE SPECTRUM IS COMPUTED
 TINTVL IS THE SAMPLING INTERVAL OF THE INPUT SERIES
 SPECTR STORES THE SPECTRUM VALUES
 MIN    IS THE ORDER OF THE A R MODEL FOR WHICH THE FPE IS MINIMUM
 FMIN IS THE MINIMUM FPE
DIMENSION X(NDATA),B1(NDATA),B2(NDATA),SPECTR(NFREQ)
DIMENSION X(600),B1(600),B2(600),SPECTR(600)
DIMENSION A(MAX),AA(MAX),AMIN(MAX),FPE(MAX),P(MAX+1)
DIMENSION A(12),AA(12),AMIN(12),FPE(12),P(13)
MAXIMUM ENTROPY POWER SPECTRUM ESTIMATIOM
AUTOREGRESSIVE MODEL FITTING
MINORD=0
FMIN=0.1E+26
PI=0.0
MAXORD=MAX
DO 10 K=1,NDATA
PI=PI+X(K)*X(K)
IF(K.EQ.NDATA) GO TO 10
B1(K)=X(K)
B2(K)=X(K+1)
CONTINUE
M=1
P(1)=PI/FLOAT(NDATA)
PNUM=0.0
PDEN=0.0
LNM=NDATA-M
DO 30 KJ=1,LNM
```

PROGRAM

```
PNUM=PNUM+B1(KJ)*B2(KJ)
PDEN=PDEN+B1(KJ)*B1(KJ)+B2(KJ)*B2(KJ)
A(M)=2.0*PNUM/PDEN
P(M+1)=P(M)*(1.0-A(M)*A(M))
IF(M.EQ.1) GO TO 50
ML=M-1
DO 40 KM=1,ML
MKM=M-KM
A(KM)=AA(KM)-A(M)*AA(MKM)
SUM=0.0
EVALUATION OF FINAL PREDICTION ERROR
IKM=M+1
DO 70 KK=IKM,NDATA
FX=X(KK)
KJKL=KK-M
RX=X(KJKL)
DO 60 JK=1,M
KJK=KK-JK
FX=FX-A(JK)*X(KJK)
JKL=KJKL+JK
RX=RX-A(JK)*X(JKL)
SUM=SUM+FX*FX+RX*RX
CONTINUE
ICB=NDATA+M+1
IDC=NDATA-M
FDATA=FLOAT(IDC)
FNUM=FLOAT(ICB)
FPE(M)=(0.5*SUM/FDATA)*FNUM/(FDATA-1.0)
CHECKING FOR MINIMUM F P E
IF(FPE(M).GT.FMIN) GO TO 90
FMIN=FPE(M)
DO 80 JMN=1,M
AMIN(JMN)=A(JMN)
MINORD=M
MIN=MINORD
```

PROGRAM

```
DO 100 KQ=1,NM
AA(KQ)=A(KQ)
JW=NDATA-M
DO 110 KS=1,JW
KQST=KS+1
B1(KS)=B1(KS)-AA(NM)*B2(KS)
B2(KS)=B2(KQST)-AA(NM)*B (KQST)
GO TO 20
POWER SPECTRUM ESTIMATION
DO 130 JM=1,MINORD
A(JM)=AMIN(JM)
FREQ=1.0/(TINTVL*FINTVL)
NFREQ=INT(FREQ)+1
DNUM=PMIN*TINTVL
PIE=3.14159265
DPOWER=2.0*PIE*TINTVL
AFMIN=(-0.5)/TINTVL
DO 150 MI=1,NFREQ
ADREAL=1.0
ADIMAG=0.0
FRQ=AFMIN+FINTVL*FLOAT(M )
AFRQ=FRQ*DPOWER
DO 140 KIM=1,MINORD
ANGL=AFRQ*FLOAT(KIM)
ADREAL=ADREAL-A(KIM)*COS ANGL)
ADIMAG=ADIMAG+A(KIM)*SIN ANGL)
CONTINUE
SDEN=ADREAL*ADREAL+ADIMAG*ADIMAG
SPECTR(MI)=DNUM/SDEN
CONTINUE
RETURN
END
```

PROGRAM

```
      SUBROUTINE MEMINT(X,NP,MP,Y)
C     THIS ALGORITHM PERFORMS INTERPCLATION BY DIRECTLY OPERATIN
C     DATA USING THE MAXIMUM ENTRCPY METHOD
C      X  IS THE INPUT DATA
C      NP  IS THE NUMBER OF THE INPUT SAMPLES. NP IS AN ODD INTE
C      MP IS THE FACTOR BY WHICH THE SAMPLING RATE IS TO BE INCR
C      Y IS THE INTERPCLATED CUTPUT SERIES
C      MATINV IS A LIBRARY SUBRCUTINE FCR INVERSION OF A MATRIX
      DIMENSICN X(80),G(12,12),D(12),C(12),GG(12,1)
      COMPLEX F(505),B(40),Z(200),TEMPO(5)
      DIMENSICN Y(80),XX(80)
C     I N T E R P O L A T I O N
      INP=(NP+1)/2
      JNP=NP/2
C     EVALUATION OF FCURIER TRANSFORM
      DC 26 J=1,INP
      GG(J,1)=0.0
      DC 26 K=1,INP
      JK=K-J+INP
   26 G(J,K)=X(JK)
      CALL MATINV(G,INP,GG,O,DETERM)
C     CCMPUTATICN  CF CCEFFICIENTS C(I),D(I)
      DC 36 JJ=1,INP
      D(JJ)=G(JJ,1)
      IF(ABS(D(1)).LT.0.001) D(1)=0.001
   36 C(JJ)=G(1,JJ)/D(1)
C     CCMPUTATICN CF CCEFFICIENTS B(I)
      DC 66 KK=1,JNP
      TEMP=0.C
      DC 76 KKK=1,KK
      MM=KKK+INP-KK
   76 TEMP=TEMP+C(MM)*D(KKK)
   66 B(KK)=CMPLX(TEMP,C.0)
      DC 86 KK=INP,NP
      TEMP=0.C
      NKK=NP+1-KK
```

PROGRAM

```
      DO 96 KKK=1,MKK
      MMM=KKK+KK-INP
96    TEMP=TEMP+C(KKK)*D(MMM)
86    B(KK)=CMPLX(TEMP,0.0)
C     TRANSFORM  EVALUATION
      TEMPO(1)=CMPLX(1.0,0.0)
      TEMPO(2)=CMPLX(0.5,0.0)
      PIE=3.14159265
      PIEX=PIE*0.004
      DO 116 IJ=1,501
      F(IJ)=CMPLX(0.0,0.0)
      ZZZ=-PIEX*FLOAT(IJ-251)
      DO 126 JK=1,NP
      ZZ=ZZZ*FLOAT(JK-INP)
      TEMPO(3)=CMPLX(0.0,ZZ)
126   F(IJ)=F(IJ)+B(JK)*CEXP(TEMPO(3))
116   F(IJ)=TEMPO(1)/F(IJ)
      DO 146 IJ=1,500
146   F(IJ)=TEMPO(2)*(F(IJ)+F(IJ+1))
C     INTERPOLATION
138   MNP=(NP-1)*MP+1
      IMNP=(MNP+1)/2
      FACT=0.5/FLOAT(MP)
      DO 155 LM=1,MNP
      Z(LM)=CMPLX(0.0,0.0)
      PIEXX=PIEX*FACT*FLOAT(LM-IMNP)
      DO 156 MN=1,500
      YYY=PIEXX*FLOAT(501-2*MN)
      TEMPO(4)=CMPLX(0.0,YYY)
156   Z(LM)=Z(LM)+F(MN)*CEXP(TEMPO(4))
      Z(LM)=Z(LM)*CMPLX(0.002,0.0)
155   Y(LM)=CABS(Z(LM))
      RETURN
      END
```

```
      SUBROUTINE OPTINT(SPECTR,N,A1,A2)
C     EVALUATION OF OPTIMUM FILTER COEFFICIENTS FOR DOUBLING THE
C     SAMPLING RATE USING FOUR NEIGHBOURING DATA POINTS
C      SPECTR IS THE SERIES OF SPECTRUM VALUES EQUISPACED ON THE
C      FREQUENCY AXIS BETWEEN ZERO AND PIE(3.14159265)
C      N IS THE NUMBER OF INPUT SPECTRUM SAMPLES
C      A1,A2  ARE THE FILTER COEFFICIENTS
      DIMENSION SPECTR(N)
      PIE=3.14159265
      F1=0.5*PIE/FLOAT(N)
      F2=3.0*F1
      S1=0.0
      S2=0.0
      S11=0.0
      S12=0.0
      S22=0.0
      DO 6060 I=1,N
      TEM=COS(F1*FLOAT(I))*SPECTR(I)
      S1=S1+TEM
      S11=S11+TEM*COS(F1*FLOAT(I))
      S12=S12+TEM*COS(F2*FLOAT(I))
      TEM=COS(F2*FLOAT(I))*SPECTR(I)
      S2=S2+TEM
 6060 S22=S22+TEM*COS(F2*FLOAT(I))
      S1=0.5*S1
      S2=0.5*S2
      SDET=S11*S22-S12*S12
      A1=(S1*S22-S2*S12)/SDET
      A2=(S2*S11-S1*S12)/SDET
      RETURN
      END
```

PROGRAM

```
SUBROUTINE FFEFTI(X,N,INVFFT)
COMPUTATION OF THE FAST FOURIER TRANSFORM
 X  IS THE COMPLEX INPUT SERIES.AFTER EXECUTION X STORES THE FFT
 N  IS THE NUMER CF DATA POINTS. N IS OF THE FORM 2**M
 INVFFT IS ASSIGNED THE VALUE 1 FOR COMPUTING INVERSE FFT
 INVFFT IS ASSIGNED ANY OTHER INTEGRAL VALUE FOR COMPUTING FFT
COMPLEX X(N),X1,X2,X3
M=0
NN=N
DO 10 I=1,N
NN=NN/2
M=M+1
IF(NN.LT.2) GO TO 20
CONTINUE
IF(INVFFT.NE.1) GO TO 40
DO 30 I=1,N
X(I)=CONJG(X(I))
N2=N/2
M1=M-1
K=0
DO 70 L=1,M
DO 60 I=1,N2
P=ISCRML(K/2**M1,M)
X1=CMPLX(0.0,-(6.28318530*P/FLOAT(N)))
K1=K+1
K1N2=K1+N2
X2=X(K1N2)*CEXP(X1)
X(K1N2)=X(K1)-X2
X(K1)=X(K1)+X2
K=K+1
K=K+N2
IF(K.LT.N) GO TO 50
K=0
M1=M1-1
N2=N2/2
DO 80 K=1,N
```

PROGRAM

```
      I=ISCRML(K-1,M)+1
      IF(I.LE.K) GO TO 80
      X3=X(K)
      X(K)=X(I)
      X(I)=X3
80    CONTINUE
      IF(INVFFT.NE.1) GO TO 100
      DO 90 K=1,N
90    X(K)=CONJG(X(K))
100   RETURN
      END
      FUNCTION ISCRML(J,M)
      J2=J
      ISCRML=0
      DO 110 I=1,M
      J1=J2/2
      ISCRML=2*ISCRML+(J2-2*J1)
110   J2=J1
      RETURN
      END
```

PROGRAM

```
SUBROUTINE WGNOIZ(X,N,M,YO,VAR)
GENERATION OF UNIFORMLY DISTRIBUTED OR GAUSSIAN WHITE NOISE
SERIES  USING MULTIPLICATIVE CONGRUENCY METHOD
 X  REPRESENTS THE SERIES
 N  IS THE NUMBER OF DATA POINTS
 YO  IS AN ODD POSITIVE INTEGER, THE INITIAL VALUE FOR RECURSION
 YO  IS AN ODD INTEGER,THE INITIAL VALUE FOR RECURSION
 VAR  IS THE DESIRED VARIANCE
 M  IS THE NUMBER OF SUCCESSIVE VALUES OF A UNIFORMLY DISTRIBUTED
 WHITE SERIES TO BE ADDED TO GET THE GAUSSIAN SERIES. IF M=1, THE
 SERIES IS UNIFORMLY DISTRIBUTED. M=12 FOR GAUSSIAN IS ADEQUATE
DIMENSION X(N),Y(50)
SIGMA=SQRT(12.0*VAR/FLOAT(M))
MOD=2
IF(N.LE.MOD) GO TO 20
MOD=2*MOD
GO TO 10
AMOD=FLOAT(4*MOD)
ALAMDA=SQRT(AMOD)
LAMDA=ALAMDA/8.0
ALAMDA=3.0+FLOAT(8*LAMDA)
Y(M+1)=YO
DO 40 J=1,N
X(J)=0.0
Y(1)=Y(M+1)
DO 30 I=1,M
Y(I+1)=Y(I)*ALAMDA
ITEMP=Y(I+1)/AMOD
Y(I+1)=Y(I+1)-FLOAT(ITEMP)*AMOD
X(J)=X(J)+Y(I)/AMOD
X(J)=SIGMA*(X(J)-0.5*FLOAT(M))
RETURN
END
```